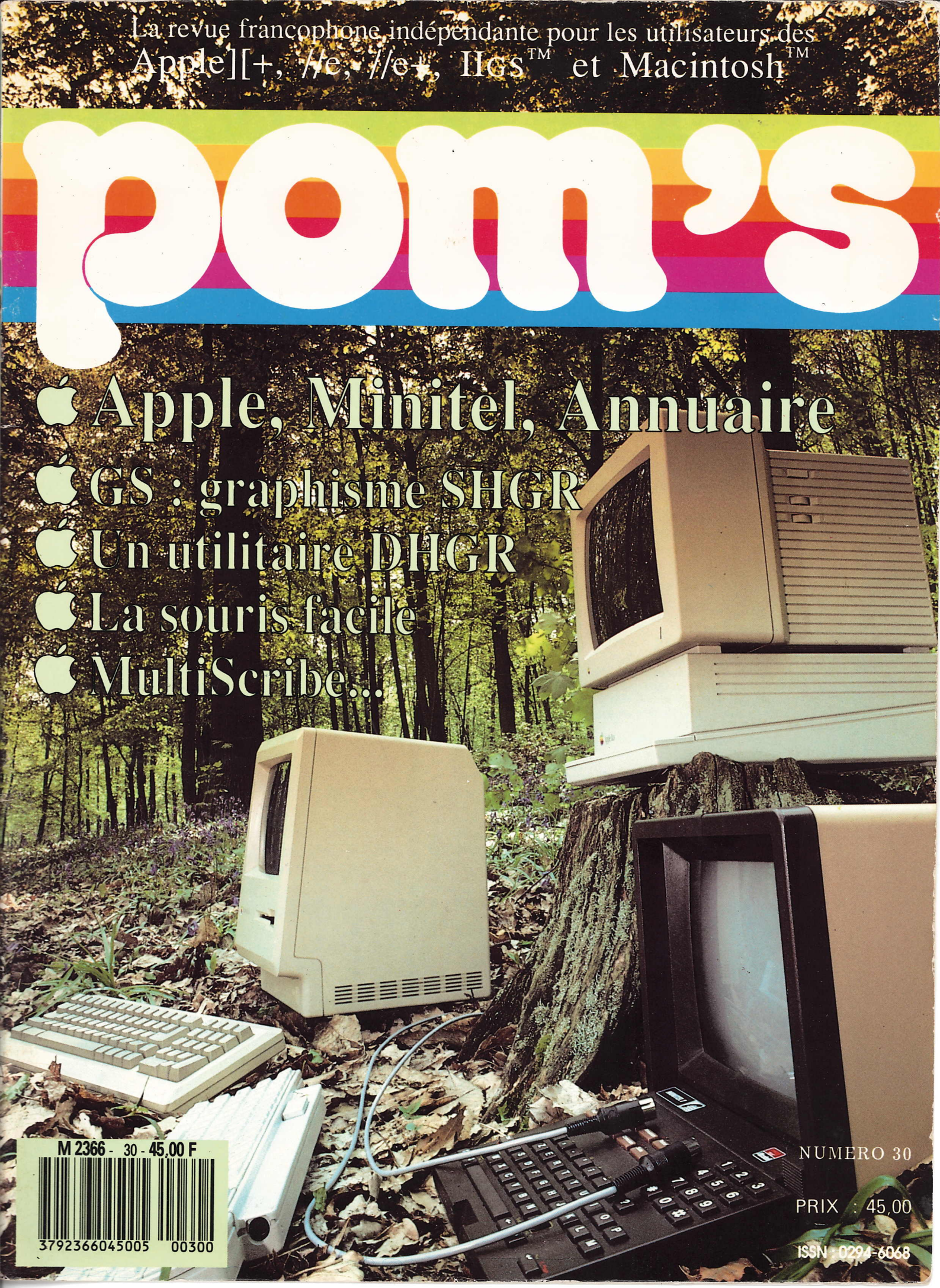
La revue francophone indépendante pour les utilisateurs des
Apple][+, //e, //e+, IIGS™ et Macintosh™

POM'S

- Apple, Minitel, Annuaire
- GS : graphisme SHGR
- Un utilitaire DHGR
- La souris facile
- MultiScribe...

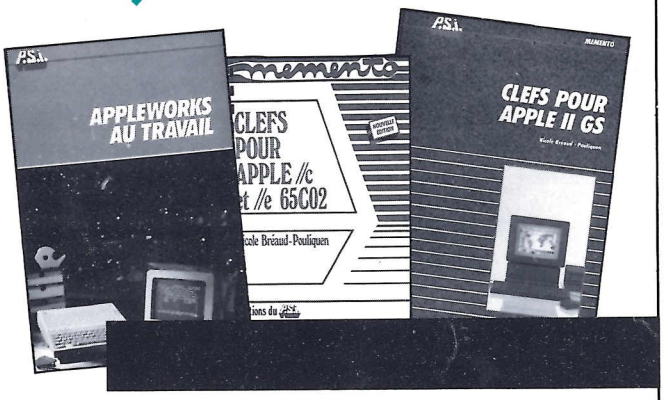


M 2366 - 30 - 45,00 F
3792366045005 00300

NUMERO 30

PRIX : 45,00

ISSN : 0294-6068



Appleworks au travail

Par A. Gargadennec et J.M. Jego - 188 pages - 160 FF

Appleworks est un logiciel intégré qui permet de résoudre les problèmes courants rencontrés en entreprise. Cet ouvrage est d'abord un livre d'apprentissage pour s'initier à Appleworks au travers d'exemples concrets. Il traite successivement des bases de données du traitement de texte et du tableau de calcul, ceci avec une difficulté croissante. Fonctionne sur Apple II e, II c et II GS.

Clefs pour Apple II c et II e 65C02

par Nicole Bréaud-Pouliquen - 180 pages - 145 FF

Ce livre est un mémento qui permet d'accéder facilement aux informations telles que les commandes Basic ou systèmes, les caractères, les messages d'erreur, les adresses et jeu d'instructions de 65 C 02. Cet ouvrage est aussi un recueil d'astuces qui permettent de justifier un texte à droite, de programmer la souris, de transférer une image en mémoire auxiliaire...
Un mémento indispensable à tout possesseurs d'Apple II et d'Apple II e 65C02.

Clefs pour Apple II GS

Deuxième édition
par Nicole Bréaud-Pouliquen - 216 pages - 250 FF

Cet ouvrage offre une synthèse des spécificités du matériel et des logiciels de développement et permet d'accéder à des informations fondamentales concernant l'organisation matérielle : mémoires et ressources graphiques, microprocesseur et jeu d'instructions, entrées/sorties... L'auteur décrit en détail les différents logiciels de développement (moniteur, système APW avec son éditeur et son micro-assembleur) et donne un répertoire de l'ensemble des outils de bureau. Un chapitre est consacré aux appels du Prodos 16. Enfin, plusieurs exemples illustrent toutes les possibilités de programmation de l'Apple II GS.

Consulter les livres P.S.I. sur Minitel au **36.15.** Tapez **OI** puis **P.S.I.**

Envoyer ce bon accompagné de votre règlement à PCV Diffusion - BP 86 - 77402 Lagny-sur-Marne Cedex

Nom _____
Prénom _____
Rue _____
Ville _____ N° _____
Code postal _____

DÉSIGNATION	PRIX
Frais de port	10,00 FF
TOTAL	

☐ Je demande le catalogue P.S.I. gratuit
☐ Paiement par chèque joint
☐ Paiement par Carte Bleue Visa
Date d'expiration _____
No _____
Signature :

POM 5

Tous Apple //

Ordico

Destinée aux amateurs de mots croisés ou de Scrabble, cette base de données, due à Roland Jost, permet de trouver un mot de longueur donnée dont on ne connaît que quelques lettres.

Ordico contient plus de 15 000 mots classés en 70 rubriques.
Recherches et affichages sont rapides : un fichier de 1 500 mots est chargé en moins de 10 secondes et exploité quasi-instantanément.
Il est bien sûr possible d'ajouter des termes aux divers fichiers, de créer de nouvelles rubriques.

Voici quelques rubriques :

1ère face :
Acteurs, Animaux, Armes/guerres, Auteurs américains, Auteurs anglais, Auteurs français, Chimie, Cinéastes, Coureurs cyclistes, Départements/régions, Dieux/déesses, Familles végétales, Femmes célèbres, Hommes politiques, Iles, Jeux/sports, Minéraux, Montagnes, Musiciens jazz, Musiciens, Parties du corps, Peintres étrangers, Peintres français, Rivières/fleuves, Saints/saintes, Savants/inventeurs, Sculpteurs, végétaux, Vêtements, Villes
2ème face :
Athlètes, Boxeurs, Cantatrices, Cols, Cosmonautes, Coureurs automobiles, Déserts, Détroits, Doctrines philosophiques, Drogues, Escrimeurs, Explorateurs, Gymnastes, Haltérophiles, Judokas, Lutteurs, Maladies, Maréchaux de France, Médicaments, Nageurs, Patineurs, Poissons, Présidents américains, Skieurs, Ski nordique, Unités, Villes olympiques

Exemples :
Un musicien dont le nom comporte 7 lettres, les 2ème et 5ème sont des 'E'. Tapez : –E– –E– –.
Vous obtenez instantanément :
BENNETT DEBOECK DELEEUW GEVAERT LESUEUR PEDRELL PEETERS WELLESZ

Dans les acteurs, –A– – – –E vous donnerait :
CARETTE PALANCE RACETTE RANDONE RAYMONE VALLONE
et – – – –H– :
CAUCHY CEECHI ENIGHT VAUGHN WRIGHT

– – – – – – – dans les femmes célèbres donnerait 32 noms...

Disquette double face et documentation 200,00 F. Bon de commande p. 74

Numéro 30
mai-juin 1987

Éditorial
Hervé Thiriez — Page 5

GSPaint et Basic
Jacques Rey — Page 6

La souris ? facile !
Alain Raynaud — Page 9

Humeur...
Page 16

MultiScribe 2.0
Essai d'un traitement de textes en DHGR
Bernard Toméno — Page 17

DhgrTool : un outil graphique
Roland Jost — Page 19

Un jeu de réflexion : Souris-Laser
Benoît Mesnil — Page 29

'patch' de BugByter
Michel Duroc — Page 37

Freeware : un billard à l'essai
Page 40

Essai Macintosh : More 2
Philippe Mathieu — Page 41

Minitel, Macintosh et le 11 TPom's
Jean-Luc Bazanegue — Page 42

Minitel, Apple // et le 11 TPom's
Christian Piard — Page 53

µ-informations
Jean-Michel Gourévitch — Page 67

Disquettes Pom's Mac B à Mac I
Page 72

Les annonceurs ; Apple : pages 38 et 39 ; Aramis : page 76 ; PSI : page 2 ; Q.S.I. : page 41.

Éditions MEV - 12, rue d'Anjou - 78000 Versailles. Tél. : (1) 39 51 24 43. Directeur de la publication : Hervé Thiriez

Les disquettes Pom's

Pour éviter les saisies fastidieuses et pas toujours fiables, Pom's met à votre disposition des disquettes d'accompagnement qui regroupent l'ensemble des programmes de la revue.

Apple //

Pour les Apple //, deux types de disquettes :

- 140 Ko, 5,25 pouces, au prix de 60,00 F, fichiers en format DOS 3.3 au recto, en format ProDOS au verso. L'ensemble des programmes de ce numéro est conçu pour fonctionner avec ProDOS ;
- 800 Ko, 3,5 pouces pour Unidisk, au prix de 80,00 F, fichiers en format ProDOS seulement.

Pom's n'a pas encore de licence d'utilisation du fichier PRODOS aussi, la face ProDOS n'est pas *bootable*. À l'aide du programme FILER ou de la disquette Utilitaires Système, installer les fichiers PRODOS et BASIC.SYSTEM sur la face ProDOS : la disquette pourra faire démarrer le système.

Macintosh

Si vous avez la disquette Mac 30, tout est simple puisque vous disposez des programmes prêts à l'emploi : ce sont des applications 'double-cliquables' et autonomes. Les fichiers 'source' autorisent des modifications – à condition de détenir le 'Système de développement 68000' – de l'application "TPom's".

Si vous n'avez pas la disquette Mac 30, et afin de pouvoir utiliser "TPom's", il vous faudra saisir les fichiers nécessaires à sa création avec l'éditeur "Edit" (Système de développement 68000), puis provoquer le traitement du fichier 'T_Pom's.Job' depuis l'application "Exec" (toujours le Système de développement 68000).

Fichiers et programmes sur la disquette Apple //

DEMO.DECOMPACT	Programme Basic de démonstration (RUN ou —)
DECOMPACT.OBJ	Routine en assembleur 65816 (BLOAD)
DECOMPACT.S	Source en format TEXT
POMS1	Image GSPaint utilisée par DEMO.DECOMPACT
POMS2 à POMS7	Idem, mais seulement sur disquettes 800 Ko
DHGRTOOL	Programme Basic à lancer par RUN ou par —
DHGRTOOL.S	Source en format TEXT
DHGRTOOL.O	Routine en asembleur
HARD	Routine de HardCopie
DEMOTOOL	Programme de démonstration des routines
IM.HGR1	Image HGR utilisé pour la démonstration
IM.HGR2	Image HGR utilisé pour la démonstration
IM.DHGR	Image DHGR utilisé pour la démonstration (800 Ko seulement)
SOURIS.LASER	Jeu Basic (RUN ou —)
LASER.START	Programme Basic de lancement du jeu en assembleur (RUN ou —)
LASER.MOUSE	1er fichier lancé par LASER.START
LASER.SBR	2ème fichier lancé par LASER.START
SOURIS.DEMO	Démonstration des routines Souris (RUN ou —)
SOURIS.C	Routine assembleur
SETCURSOR.C	Routine assembleur
INTERFACE.C	Routine assembleur
INVERSION.C	Routine assembleur
SOURIS	Source assembleur en format TEXT
SETCURSOR	Source assembleur en format TEXT
INTERFACE	Source assembleur en format TEXT
INVERSION	Source assembleur en format TEXT
T.POMS	Programme objet (BRUN ou —)
T.POMS1 à T.POMS4	Sources en format TEXT

Fichiers et programmes sur la disquette Macintosh

Girl	Image MacPaint
Japanese Girl	Image MacPaint
Screen Editor	Application
T_Pom's	Application
T_Pom's.Asm	Fichier d'assemblage MDS 68000 (Assembler)
T_Pom's.Job	Fichier 'exec' MDS 68000 (Executive)
T_Pom's.Link	Fichier de 'linkage' MDS 68000 (Linker)
T_Pom's.R	Fichier de ressources MDS 68000 (RMaker)
T_Pom's/1.Asm	Fichier d'assemblage MDS 68000 (Assembler)
T_Pom's/2.Asm	Fichier d'assemblage MDS 68000 (Assembler)
T_Pom's/3.Asm	Fichier d'assemblage MDS 68000 (Assembler)
T_Pom's/4.Asm	Fichier d'assemblage MDS 68000 (Assembler)

Hervé Thiriez

Editorial

Voici donc bouclé le 30ème numéro de Pom's, numéro rajeuni par le printemps mais également par notre souci de qualité, y compris dans la présentation. Pom's évolue, s'ouvre au GS, à la communication, à la télématique, bientôt au Mac SE et au Mac II ; ProDOS se doit de devenir la règle — la diversité des supports nous y contraint et c'est un bonne contrainte —, DOS 3.3 est en passe de devenir exception. C'est le sens de l'évolution Apple, MEM/DOS s'appuie d'ailleurs déjà sur ProDOS qui est un pont avec les MFS et HFS du Mac.
Le IIGS lui-même est un pont avec la ligne Mac : l'acquéreur du nouveau-né, plongé dans la volumineuse documentation distribuée par VIF ou dans le livre de J.-P. Curcio au PSI sera familier de la ToolBox et des routines QuickDraw de son appareil... et de celles du Mac.

Dans ces pages, télématique toujours : après l'enregistrement des pages des serveurs Minitel, le téléchargement entre Apple via Minitel et téléphone, voici des programmes de récupération de l'annuaire électronique, sur Apple // et sur Macintosh. Votre câble de liaison ne chômera pas : votre Apple // et votre Mac *serviront* bientôt...
La souris est à l'honneur : une routine pour en faciliter la gestion depuis le Basic — l'Applesoft est bien pauvre là encore — et un double jeu de réflexion construit autour de la souris, version Basic, version assembleur 65C02 (à vos kits...).
Vous lirez également des bancs d'essai, MultiScribe, More 2 et la passion de nos rédacteurs : le billard. Quant à GSPaint et le Basic, il s'agit bien là de la première routine 65816 dans Pom's : ne doutez pas de l'extension de cette rubrique...

Ont collaboré à ce numéro : Alexandre Avrane, Jean-Luc Bazanegue, Michel Duroc, Jean-Michel Gourévitch, Olivier Herz, Roland Jost, Philippe Mathieu, Benoît Mesnil, Gérard Michel, Christian Piard, Alain Raynaud, Jacques Rey, Hervé Thiriez, Bernard Toméno.
Directeur de la publication, rédacteur en chef : Hervé Thiriez.
Rédacteurs : Alexandre Avrane, Olivier Herz.
Siège social : Éditions MEV - 12, rue d'Anjou - 78000 Versailles. Tél. : (1) 39.51.24.43.
Publicité : Éditions MEV.
Diffusion : N.M.P.P.
Impression : Berger-Levrault - 18, rue des Glacis - 54000 Nancy. Tél. : 83.35.61.44.
Photo de couverture : CP & JLB.

Pom's est une revue indépendante non rattachée à Apple Computer, Inc. ni à Apple Computer France S.A.R.L. Apple, le logo Apple, Mac et le logo Macintosh sont des marques déposées d'Apple Computer, Inc.

GSPaint et Basic

Jacques Rey

Après avoir craqué pour un superbe IIGS, il faut se rendre à l'évidence, que faire sans l'ombre d'une documentation dans le carton, sans un seul renseignement sur Prodos 16, sur le port imprimante (que refusent obstinément certains programmes), sans parler de la synthèse sonore qui avait charmé nos oreilles dans la boutique ? Il n'y a que 'Clefs pour Apple IIGS' qui contient beaucoup de choses encore bien vagues... (La documentation technique maintenant disponible, 3000 pages en 6 classeurs, 2000,00 F, peut rendre le service espéré).

Nous disposons, bien sûr, de GS/Paint, c'est beau et spectaculaire, mais que faire de ces images que l'on ne peut manipuler qu'avec le programme ? Un fichier de 32 Ko, en format écran, ne se manipule pas facilement sous environnement Basic.system. Aussi était-il tentant de charger en mémoire les images sous format GS/Paint et de les transférer en page Super Haute Résolution à l'aide d'une routine de décompactage.

Structure d'une image GS

La page Super HGR se trouve dans le banc de mémoire $E1 ($00 étant le banc accessible par Basic.sytem et Prodos 8), elle débute en $2000 pour finir en $9FFF. L'image en elle-même est constituée de 200 fois 160 octets situés entre $2000 et $9CFF, chaque demi-octet représentant 1 pixel (en mode 320 x 200 adopté par GS/Paint) qui peut prendre 16 valeurs définissant le numéro de rang d'une des couleurs d'une palette.
Cette palette est formée d'une suite de 16 fois deux octets qui par leurs combinaisons de valeurs constituent l'une des 4096 nuances, l'emplacement réservé commence en $9E00 et se termine en $9FFF ce qui permet de loger 16 palettes différentes !

De $9D00 à $9DC7 se trouvent 200 octets qui contrôlent chacun une ligne du dessin ; la partie basse de ces octets, en particulier, désigne la palette qui est active pour une ligne donnée ce qui permet d'afficher 256 nuances dans une page. Dans le cas de GS/Paint une seule palette est active, ce qui simplifie bien les choses.

Analysons maintenant la structure d'un fichier image compactée par GS/Paint.

On reconnaît facilement que les 32 premiers octets représentent la palette de couleurs, viennent ensuite 514 octets qui déroutent quelque peu et qui constituent finalement les canevas disponibles sur la ligne inférieure de l'écran de GS/Paint. Ces octets ne seront d'aucune utilité pour la transcription du dessin.
L'image proprement dite commence à partir du $222ème octet du fichier.

Pour compacter une image, il faut isoler les groupes de pixels identiques des suites de points différents les uns des autres. Le fichier compacté sera alors constitué de pointeurs qui indiquent si l'on doit multiplier x fois l'octet suivant, ou bien recopier les x octets qui suivent directement.

Le décompactage

Dans le cas de GS/Paint, la valeur donnée au pointeur va prendre trois significations différentes :

- Un pointeur inférieur à $40 indique que nous avons à faire à un segment contenant des octets non identiques, le pointeur donne la longueur du segment.
- Les valeurs comprises entre $C0 et $ E7 ainsi que $41, $42, $44, $45 et $46 nous invitent à multiplier l'octet suivant de 2 à 160 fois : pour cela, $C0 vaut coefficient 4, $C1 vaut 8, $C2 = 12 et ainsi de suite de 4 en 4 jusqu'à $E7 = 160; les coefficients x2, x3, x5, x6 et x7 étant représentés par les valeurs $41 à $46.
- Si le pointeur est compris entre $80 et $C0, il faut prendre en compte les quatre octets qui suivent et reproduire cette combinaison x fois. La valeur de x étant trouvée en retranchant $80 du contenu du pointeur.

L'exemple en encadré éclaircira le codage.

La routine de décompactage est courte (moins de 256 octets), elle a été assemblée en $2000 en utilisant le Mini-moniteur, faute de mieux, le fichier 'source' listé ci-contre et figurant sur la disquette d'accompagnement est un fichier texte créé sur Appleworks en respectant la syntaxe approximative d'un assembleur. Il est récupérable avec tout assembleur.

Le but était avant tout d'utiliser des instructions spécifiques du processeur 65C816, les transferts de blocs d'octets étant effectués à l'aide de l'instruction MVN qui copie A octets de l'adresse X vers l'adresse Y (X et Y sur 16 bits). Les bancs de mémoire source et destination sont indiqués par les deux octets qui suivent l'instruction MVN.

Il faut signaler deux contraintes importantes à l'utilisation de MVN :

- le registre B (registre indiquant le banc de donnée actif) conserve la valeur du banc de destination en fin d'instruction, il est donc nécessaire de le sauvegarder par PHB (empiler B) avant MVN et le restituer par PLB après ;
- si l'on travaille en mode natif mixte (A sur 8 bits, X et Y en 16 bits), ce qui est le cas de la routine, il faut impérativement annuler l'octet haut du registre A car l'instruction MVN décompte tout de même sur 16 bits (utiliser LDA £00 puis XBA avant de charger dans A le nombre d'octets à transférer).

La routine comportant quelques JSR internes (deux sous-routines de décomptage) ne peut être relogée qu'après des modifications très simples à réaliser. Pour lister le code objet en moniteur, ne pas oublier de taper au préalable 0=x pour que le désassemblage se fasse sans erreurs et reconnaisse les codes sur 16 bits (la lettre x doit être tapée en minuscule).

L'adresse de chargement du fichier image compacté est $2100, pour la changer, modifier la valeur du 13ème octet.

Le deuxième octet contient la valeur qui permet la commutation du registre $C029 en mode Super HGR. Si l'on met $41 l'image est écrite mais l'écran reste en mode texte, il faut alors taper POKE 49193,161 pour l'afficher et POKE 49193,65 pour revenir au texte ($C029:A1 et $C029:41 sous moniteur).
En plaçant $A1 dans le deuxième octet, le décompactage a lieu en direct ce qui peut créer un certain effet d'animation.

Lorsqu'on se trouve en Super HGR, pour revenir au mode texte depuis le basic, seul le poke cité plus haut est actif. Par contre, sous moniteur, taper CTRL-T puis Return.

Le programme DEMO.DECOMPACT illustre l'utilisation de la routine.

Source DECOMPACT.S

JIgs

```
*
* Version  1.4 - J. REY   Février/Mars 1987    *
*
        ORG $2000
*
ADSOU   =   $EB      ;Pointe sur la zone de chargement de l'image compactée
ADDES   =   $ED      ;Pointe sur la zone de la page super HGR
COMP1   =   $EF      ;Compteur d'octets
COMP2   =   $FA      ;Compteur de transfert
VSR     =   $C029    ;Vidéo Sélect Register : commutateur de résolution vidéo
*
*************** Initialisation ***************
*
INIT    LDA £41      ;Ou LDA £A1 pour voir l'image se décompacter
        STA VSR      ;Permet la linéarisation des adresses super HGR
        STZ ADSOU    ;Initialisation des adresses source et
        STZ ADDES    ;destination
        LDA £21      ;Image compactée chargée à partir de $2100
        STA ADSOU+1
        LDA £20      ;Début d'image super HGR en $E1/2000
        STA ADDES+1
        CLC
        XCE          ;Passage en mode natif 65C816
        REP £10      ;Adoption du mode mixte A = 8 bits, X et Y = 16 bits
*
*************** Transfert de la palette ****************
*
TRANSPAL PHB         ;Sauvegarde sur la pile du N° de banc de données actuel
        LDA £00
        XBA          ;Annule les 8 bits de poids fort de A
        LDA £1F      ;La palette comporte 32 octets
        LDX ADSOU    ;Récupération de l'adresse source
        LDY £9E00    ;Init de l'adresse de destination de la palette
        MVN 00E1     ;Transfert de A octets du banc 00 vers E1
        LDX £00C7
 RAZ    STZ 9D00,X   ;Remise à zéro de la zone Scan Line Control Bloc
        DEX          ;200 octets
        BPL RAZ
        PLB          ;On se replace dans le banc 00 (B = E1 après MVN)
*
*************** Analyse de l'image compactée ****************
*
DEBIMAG LDA £22
        STA ADSOU
        CLC
        LDA ADSOU+1
        ADC £02
        STA ADSOU+1  ;Premier pointeur d'image situé au $222 eme octet
*

*
LECPOINT LDA (ADSOU) ;Récupère un pointeur
        CMP £40
        BCC TRANSOCT ;Transfert d'une segment d'octets non identiques
        CMP £C0
        BCS SUPC0    ;Le pointeur est un multiplicateur de 4 en 4
        CMP £80
        BCS TRANS4   ;Transfert d'une suite de 4 octets reproduite plusieurs fois
*
OCTMULT AND £0F      ;Le pointeur est un multiplicateur par 2,3,5,6 ou 7
        INC          ;(valeur comprise entre $41 et $46)
        BRA SUITE
SUPC0   SEC          ;OCTMULT commence ici si pointeur >= à $C0
        SBC £3F      ;Transforme les valeurs du pointeur comprises
        ASL          ;entre C0 et E7 en une valeur de multiplication
        ASL          ;d'octets comprise entre 4 et 160 (de 4 en 4)
SUITE   STA COMP1    ;Initialise le compteur
        JSR INCADSOU ;Incrémente l'adresse source
LECPIX  LDA (ADSOU)  ;Récupère la valeur de l'octet 'Pixel' à multiplier
        LDY £00E1
        PHY          ;Sauve sur la pile le N° de banc source et destination
        PLB          ;Commute sur le banc destination
        STA (ADDES)  ;Ecrit dans la page Graphique
        PLB          ;Commute sur le banc source
        JSR INCADDES ;Incrémente l'adresse de destination
        DEC COMP1
        BNE LECPIX   ;On recommence autant que nécessaire
        JSR INCADSOU ;Passe au pointeur suivant
        BRA LECPOINT ;Branche au début de l'analyse
```

```
*
TRANS4   SEC          ;Début routine de multiplication de groupes de 4 octets
         SBC £80      ;Le mutiplicateur va de $0 à $A7 (1 à 40 fois 4 octets)
         STA COMP2    ;Que l'on range dans le compteur de transfert
         LDA £03      ;Transfert d'une suite de 4 octets
TRANSOCT STA COMP1
         JSR INCADSOU ;Pour se placer sur le premier octet de la suite
ENCORE   PHB          ;Sauve le banc source
         LDA £00
         XBA          ;Annule l'octet haut de A
         LDA COMP1    ;Nombre d'octets à transférer
         LDX ADSOU    ;Venant du banc 00
         LDY ADDES    ;Vers banc E1
         MVN 00E1     ;Et hop ! c'est transféré !
         PLB          ;Retour au banc de départ
 BOUCLE  JSR INCADDES ;On avance de deux Pixels (un octet)
         LDA COMP2
         BNE SAUT     ;Si transferts multiples saute incrémentation de
*                      l'adresse source
         JSR ADSOU    ;Incrémente l'adresse source
SAUT     DEC COMP1
         BPL BOUCLE   ;Avance les adresses du nombre d'octet transféré
         LDA £03
         STA COMP1
         DEC COMP2
         BPL ENCORE   ;A nouveau les quatre mêmes octets
         STZ COMP2
         BRA LECPOINT ;C'est pas fini, aux suivants !
*
*************** Sous routines ****************
*
INCADSOU INC ADSOU    ;Incrémente la partie basse et haute de l'adresse
         BNE RET1     ;source
         INC ADSOU+1
RET1     RTS
*
INCADDES INC ADDES    ;Incrémente l'adresse des pixels dans la page
         BNE RET2     ;écran super HGR
         INC ADDES+1
         LDA ADDES+1
         CMP £9D      ;Vérifie si l'on n'est pas arrivé à la fin de la page
         BNE RET2     ;Non ? et bien continuons !
*
FIN      PLX          ;C'est fini, on dépile pour récupérer l'adresse de retour
         SEC
         XCE          ;Retour au mode émulation
RET2     RTS          ;Au revoir !
```

Récapitulation DECOMPACT.OBJ

Après avoir saisi cette récapitulation sous moniteur, vous la sauvegarderez par : BSAVE DECOMPACT.OBJ,A$1000,L$101

```
1000:A9 A1 8D 29 C0 64 EB 64
1008:ED 64 FA A9 21 85 EC A9
1010:20 85 EE 18 FB C2 10 8B
1018:A9 00 EB A9 1F A6 EB A0
1020:00 9E 54 E1 00 A2 C7 00
1028:9E 00 9D CA 10 FA AB A9
1030:22 85 EB 18 A5 EC 69 02
1038:85 EC B2 EB C9 40 90 34
1040:C9 C0 B0 09 C9 80 B0 25
1048:29 0F 1A 80 05 38 E9 3F
1050:0A 0A 85 EF 20 A1 20 B2
1058:EB A0 E1 00 5A AB 92 ED
1060:AB 20 A8 20 C6 EF D0 EF
1068:20 A1 20 80 CD 38 E9 80
1070:85 FA A9 03 85 EF 20 A1
1078:20 8B A9 00 EB A5 EF A6
1080:EB A4 ED 54 E1 00 AB 20
1088:A8 20 A5 FA D0 03 20 A1
1090:20 C6 EF 10 F2 A9 03 85
1098:EF C6 FA 10 DC 64 FA 80
10A0:99 E6 EB D0 02 E6 EC 60
10A8:E6 ED D0 0B E6 EE A5 EE
10B0:C9 9D D0 03 FA 38 FB 60
10B8:00 00 00 00 00 00 00 00
10C0:D2 CF D5 D4 C9 CE C5 A0
10C8:C4 C5 C3 CF CD D0 C1 C3
10D0:AA AA AA AA AA AA AA AA
10D8:AA AA AA AA AA AA AA AA
10E0:A8 C3 A9 A0 CA AE A0 D2
10E8:C5 D9 A0 B0 B2 AF B8 B7
10F0:AA AA AA AA AA AA AA AA
10F8:AA AA AA AA AA AA AA AA
1100:00
```

ProDOS

Programme DEMO.DECOMPACT

Pour utiliser ce programme de démonstration, vous devez avoir sur la disquette 7 images graphiques type GS/Paint, sauvegardées sous les nom : POMS1, POMS2, POMS3... POMS7. C'est le cas si vous disposez la disquette d'accompagnement 3,5 pouces, 800Ko de Pom's.

Selon le nombre d'images disponibles, modifier la ligne 140.

```
 5  REM   ********** Démo Décompact
 10  TEXT : HOME
 20 D$ =  CHR$ (4):TYPE$ = ",T$C0,A$2100"
 25  PRINT D$"bloaddecompact.obj,a$2000"
 60  PRINT : PRINT "      Décompacteur d'image
     s GS/Paint"
 70  PRINT "      ****************************
     **"
 80  PRINT : PRINT "      Par J. Rey    * Févri
     er/Mars 87
 85  SPEED= 100: PRINT : PRINT
 90  PRINT "Nous allons voir apparaître une s
     uite    "
100  PRINT "d'images créées avec GS/Paint et
     cela    "
110  PRINT "sans l'aide du programme de dess
     in.     "
120  PRINT "La routine 'Décompact' permet d'
     utiliser"
130  PRINT "ces dessins à partir d'un progra
     mme en  ": PRINT "BASIC."
135  PRINT D$"bloadpoms1";TYPE$: CALL 8192:
     POKE 8193,161: POKE 49193,161: FOR T = 1
      TO 2000: NEXT
140  FOR X = 2 TO 7
150  PRINT D$"bloadPOMS"X;TYPE$: CALL 8192:
     NEXT
230  FOR X = 1 TO 5000: NEXT : HTAB 1: VTAB
     6: CALL  - 958: POKE 49193,65: SPEED= 25
     5
240  PRINT : PRINT : PRINT : PRINT : PRINT :
      PRINT "Pour recommencer presser 'Espace
     '": PRINT : PRINT "Pour quitter presser
     'ESC'            ";: GET R$
250  IF R$ =  CHR$ (27) THEN  HOME : END
260  GOTO 135
```

La souris ? facile !

Alain Raynaud

Pour gérer la souris, Apple propose actuellement deux méthodes :
- en assembleur, en appelant des routines placées sur la ROM de la carte interface,
- en Applesoft, avec des séries complexes de PR#4 : PRINT CHR$(0) : PR#0...

Aucune n'est totalement satisfaisante car elles exigent, de la part du programmeur, soit une parfaite connaissance du 6502, soit des acrobaties pour pouvoir gérer en Basic le moindre menu déroulant.

Le programme proposé ici offre la possibilité d'utiliser en Applesoft des commandes structurées qui permettent la programmation rapide d'applications complexes utilisant la souris.

Les commandes

Ces commandes utilisent l'éternel vecteur de l'ampersand – ou esperluète pour les francophiles anglophobes – comme point d'entrée :

& INIT (MOUSE)
débute obligatoirement toute série d'instructions utilisant la souris, et remet à zéro les pointeurs

& CLOSE (MOUSE)
déconnecte la souris à la fin du programme

& CURS (n)
sélectionne la forme du curseur graphique matérialisant la position de la souris

& WAIT (PRESS)
attend que le bouton soit appuyé pour rendre la main au programme, et continue d'animer le curseur en attendant

& WAIT (NOPRESS)
idem mais attend que le bouton soit relâché

& WAIT (NEW PRESS)
idem mais attend une nouvelle pression du bouton

& WAIT (OUT [0,0 TO 100,100])
attend une sortie du cadre graphique défini par les deux points de coordonnées 0,0 et 100,100

& WAIT (OUT [0,0 TO 100,100] OR PRESS)
idem mais on rend immédiatement la main si le bouton est activé

& WAIT (OUT [0,0 TO 100,100] OR NOPRESS)
idem mais on rend immédiatement la main si le bouton est relâché

& INVERSE (0,0 : LEN = 10,6)
inverse la fenêtre graphique déterminée par le point 0,0 sur une longueur de 10 octets (soit 10 x 7 = 70 pixels) et sur une hauteur de 6 lignes

& DATA (X,Y,BUT)
place respectivement dans les variables X, Y et BUT : l'abscisse, l'ordonnée de la souris et l'état du bouton (0=relâché, 1=appuyé)

Exécution du programme

Le programme se charge par la séquence d'instructions suivante :

```
BLOAD SOURIS.C
BLOAD SETCURSOR.C
BRUN  INTERFACE.C
BLOAD INVERSION.C
```

suivi des commandes:

```
HGR : POKE -16302,0
```

pour afficher la première page graphique en mode non-mixte.
Il est recommandé de placer HIMEM: à 36608 ($8F00) avant le chargement de ces routines afin d'éviter leur écrasement intempestif.

L'ensemble des routines fonctionne sans modification sous environnement DOS 3.3 ou ProDOS. Les sources ont été constitués avec l'assembleur ProCODE.

En revanche, pour minimiser l'occupation mémoire du programme, aucun contrôle n'est effectué pour vérifier qu'il existe bien une souris en slot 4, si tel n'est pas le cas alors gare au *plantage* !

Occupation mémoire

$9000
début du fichier SOURIS,
initialisation de la souris.
appel de la routine en ROM correspondant au code d'appel

$9200
début du fichier INTERFACE.
interprétation des commandes Basic et appel des routines des autres fichiers

$94F0
début du fichier SETCURSOR.
contient la table de formes (SHAPE) Applesoft matérialisant la souris

$95C0
début du fichier INVERSION, routine d'inversion de fenêtre graphique

$9600
début du DOS

Sept formes sont définies dans le fichier SETCURSOR :

0 flèche
1 ciseau
2 note de musique
3 croix
4 stylo
5 crayon
6 document
7 feuille

Il vous est possible d'en définir d'autres sous la contrainte d'espace mémoire indiquée ci-dessus. SETCURSOR place également aux adresses 2 et 3 la position de la forme sélectionnée.

Programme SOURIS.DEMO

```
5   PRINT CHR$ (4)"BLOAD SETCURSOR.C"
6   PRINT CHR$ (4)"BLOAD SOURIS.C"
7   PRINT CHR$ (4)"BRUN INTERFACE.C"
8   PRINT CHR$ (4)"BLOAD INVERSION.C"
9   HGR : POKE  - 16302,0
10   & INIT(MOUSE)
20   & CURS( INT ( RND (1) * 7 + 1))
30   &  INVERSE (0,0: LEN  = 20,191)
40   & CURS( INT ( RND (1) * 7 + 1))
50   &  WAIT (OUT[0,0 TO 140,191] OR PRESS)
60   &  DATA (X,Y,BUT)
70   IF BUT > 0 THEN 200
80   &  INVERSE (0,0: LEN  = 40,191)
90   & CURS( INT ( RND (1) * 7 + 1))
100   &  WAIT (OUT[140,0 TO 279,191] OR
      PRESS)
110   &  DATA (X,Y,BUT)
120   IF BUT > 0 THEN 200
130   &  INVERSE (0,0: LEN  = 40,191)
140   GOTO 40
200   & CLOSE(MOUSE)
210    TEXT
```

Source INTERFACE Assembleur ProCODE

```
*********************************
*                               *
* programme d'interfacage de la *
*                               *
* routine : SOURIS avec le basic *
*                               *
* instructions :                *
*                               ***************
*          &INIT(MOUSE)                        *
*          &WAIT(PRESS)                        *
*          &WAIT(NOPRESS)                      *
*          &WAIT(OUT[0,0 TO 1,1])              *
*          &WAIT(OUT[0,0 TO 1,1] OR PRESS)     *
*          &WAIT(OUT[0,0 TO 1,1] OR NOPRESS)   *
*          &WAIT(NEW PRESS)                    *
*          &CURS(3)                            *
*          &CLOSE(MOUSE)                       *
*          &DATA(X,Y,BUT)                      *
*          &INVERSE(3,5:LEN-10,4)              *
*                                              *
************************************************
```

```
        ORG  $9200        ;suite de SOURIS
SOURIS  EQU  $901C

AMPER   LDA  #$4C
        STA  $3F5
        LDA  #<DEBUT
        STA  $3F6
        LDA  #>DEBUT
        STA  $3F7
        RTS

DEBUT   CMP  #$B5         ;code de WAIT
        BEQ  WAIT1
        CMP  #'I'
        BEQ  INIT
        CMP  #'C'
        BEQ  CLOSE1
        CMP  #$83         ; code de DATA
        BEQ  DATA1
        CMP  #$9E
        BEQ  INV
SNTX    RTS
WAIT1   JMP  WAIT
DATA1   JMP  DATA
CLOSE1  JMP  CLOSE

INIT    LDY  #>IN
        LDX  #<IN
        JSR  VERIF
        BCS  SNTX
        JSR  $9000
        JMP  $B1
INV     JSR  $B1
        CMP  #'('
        BNE  SNTX
        JSR  $B1
        JSR  $DD67
        JSR  $E10C
        LDA  $A1
        STA  TAB
        JSR  $B1
        JSR  $DD67
        JSR  $E10C
        LDA  $A1
        STA  TAB+1
        JSR  $B7
        CMP  #':'
        BNE  SNTX
        JSR  $B1
        CMP  #$E4         ; code de LEN
        BNE  SNTX
        JSR  $B1
        CMP  #$D0
        BNE  SNTX
        JSR  $B1
        JSR  $DD67
        JSR  $E10C
        LDA  $A1
        STA  TAB+2
        JSR  $B1
        JSR  $DD67
        JSR  $E10C
        LDA  $A1
        STA  TAB+3
        LDA  #>TAB
        STA  $1
        LDA  #<TAB
        STA  $0
        JSR  $95C0        ; INVERSE
        JMP  $B1

SNTX5   RTS

CLOSE   JSR  $B1
        CMP  #'U'
        BEQ  CUR
        LDY  #>CL         ;& CLOSE(MOUSE)
        LDX  #<CL
        JSR  VERIF
        BCS  SNTX5
        LDA  $C412        ;(SetMouse)
```

```
        STA  ]1+1
        LDA  #$00
        LDX  #$C4
        LDY  #$40
]1      JSR  $C400
        JMP  $B1

CUR     JSR  $B1
        CMP  #'R'
        BNE  SNTX5
        JSR  $B1
        CMP  #'S'
        BNE  SNTX5
        JSR  $B1
        CMP  #'('
        JSR  $B1          ; & CURS(n)
        JSR  $DD67
        JSR  $E10C
        LDA  $A0
        BNE  SNTX4
        LDA  $A1
        JSR  $94E7
        JSR  $B1
        CMP  #')'
        BNE  SNTX4
        JSR  $B1
SNTX4   RTS
DATA    JSR  $B1          ; & DATA (X,Y,BUT)
        CMP  #'('
        BNE  SNTX4
        JSR  $B1
        JSR  $DFE3
        LDA  $83
        STA  TAMP1
        LDA  $84
        STA  TAMP2
        JSR  $B1
        JSR  $DFE3
        LDA  $83
        STA  TAB
        LDA  $84
        STA  TAB+1
        JSR  $B1
        JSR  $DFE3
        LDA  $83
        STA  TAB+2
        LDA  $84
        STA  TAB+3
        JSR  $B7
        CMP  #')'
        BNE  SNTX4
        LDA  #6
        JSR  SOURIS
        LDY  $8
        JSR  $E301
        LDX  TAB
        LDY  TAB+1
        JSR  $EB2B
        LDA  $7
        LDY  $6
        JSR  $E2F2
        LDX  TAMP1
        LDY  TAMP2
        JSR  $EB2B
        LDA  $77C
        AND  #$80
        TAY
        JSR  $E301
        LDX  TAB+2
        LDY  TAB+3
        JSR  $EB2B
        JMP  $B1
SNTX2   RTS

WAIT    JSR  $B1
        CMP  #'('
        BNE  SNTX2
        JSR  $B1
        LDX  $B8
        STX  TAMP1
        LDX  $B9
        STX  TAMP2
        LDY  #>PR         ; & WAIT(PRESS)
        LDX  #<PR
        JSR  VERIF
        BCS  S1
        LDA  #0
        JSR  SOURIS
        JMP  $B1
S1      LDX  TAMP1        ; & WAIT(NOPRESS)
        STX  $B8
        LDX  TAMP2
        STX  $B9
        LDY  #>NO
        LDX  #<NO
        JSR  VERIF
        BCS  S2
        LDA  #1
        JSR  SOURIS
        JMP  $B1
S2      LDX  TAMP1        ; & WAIT(NEW PRESS)
        STX  $B8
        LDX  TAMP2
        STX  $B9
        JSR  $B7
        CMP  #$BF         ; code de NEW
        BNE  S3
        JSR  $B1
        LDY  #>PR
        LDX  #<PR
        JSR  VERIF
        BCS  SNTX2
        LDA  #5
        JSR  SOURIS
        JMP  $B1
SNTX3   RTS
S3      LDX  TAMP1
        STX  $B8
        LDX  TAMP2
        STX  $B9
        LDY  #>OU         ; & WAIT(OUT[0,0 TO 140,90])
        LDX  #<OU
        JSR  VERIF
        BNE  SNTX2
        JSR  $B1
        JSR  $DD67
        JSR  $E10C
        LDA  $A1
        STA  TAB
        LDA  $A0
        STA  TAB+1
        JSR  $B7
        CMP  #','
        BNE  SNTX3
        JSR  $B1
        JSR  $DD67
        JSR  $E10C
        LDA  $A1
        STA  TAB+2
        JSR  $B7
        CMP  #$C1         ; code de TO
        BNE  SNTX1
        JSR  $B1
        JSR  $DD67
        JSR  $E10C
        LDA  $A1
        STA  TAB+3
        LDA  $A0
        STA  TAB+4
        JSR  $B7
        CMP  #','
        BNE  SNTX1
        JSR  $B1
        JSR  $DD67
        JSR  $E10C
        LDA  $A1
        STA  TAB+5
        LDA  #<TAB
        STA  $0
        LDA  #>TAB
        STA  $1
        JSR  $B7
        CMP  #']'
        BNE  SNTX1
        JSR  $B1
```

```
        CMP  #')'
        BEQ  CH1
        CMP  #$CE        ; code de OR
        BNE  SNTX1
        JSR  $B1
        LDX  $B8
        STX  TAMP1
        LDX  $B9
        STX  TAMP2
        LDX  #<PR
        LDY  #>PR
        JSR  VERIF
        BCC  CH2
        LDX  TAMP1
        STX  $B8
        LDX  TAMP2
        STX  $B9
        LDX  #<NO
        LDY  #>NO
        JSR  VERIF
        BCS  SNTX1
CH3     LDA  #2          ; ou NOPRESS
        JMP  CALL
CH1     LDA  #4          ; simplement ')'
        JMP  CALL
CH2     LDA  #3          ; ou PRESS
CALL    JSR  SOURIS
        JMP  $B1
SNTX1   RTS

VERIF   STX  AD+1
        STY  AD+2
        STX  V1+1
        STY  V1+2
        LDX  #0
        JSR  $B7
AD      CMP  $200,X
        BNE  FAUX
        INX
V1      LDA  $200,X
        BEQ  OK
        JSR  $B1
        JMP  AD
OK      CLC
        RTS
FAUX    SEC
        RTS

TAMP1   HEX  00
TAMP2   HEX  00

IN      ASC  'INIT(MOUSE)'
        HEX  00
CL      ASC  'LOSE(MOUSE)'
        HEX  00
NO      ASC  'NO'
PR      ASC  'PRESS)'
        HEX  00
OU      ASC  'OUT('
        HEX  00
TAB     DS   6
```

Source SETCURSOR Assembleur ProCODE

```
*********************************************************
*                                                       *
*          Programme de gestion du curseur              *
*                                                       *
*********************************************************

        ORG  $94E7

AD1     EQU  $2
AD2     EQU  AD1+1

*************************
*                       *
* 00 : fleche           *
* 01 : ciseaux          *
* 02 : note de musique  *
* 03 : croix            *
* 04 : stylo            *
* 05 : crayon           *
* 06 : document         *
* 07 : feuille          *
*                       *
*************************

CHOIX   ASL              ; multiplication par 2
        TAX
        LDY  T1+1,X
        LDA  T1,X
        STA  AD2
        STY  AD1
        RTS

T1      DDB  FLECHE
        DDB  CISEAUX
        DDB  NOTE
        DDB  CROIX
        DDB  STYLO
        DDB  CRAYON
        DDB  DOC
        DDB  FEUILLE

FLECHE  HEX  2D2D1E3F372D353F374D0E0E2527270600
CISEAUX HEX  2E2E2E2E2E0EAE65E4FF1B16F6E7646D4043432B26
        HEX  2017240600
NOTE    HEX  2E2D0C3F272D252424242424240E0E2E3E2E1E3600
CROIX   HEX  20B40A2DD73326C03F0500
STYLO   HEX  49E40DFC1C2D2D2424242424242424242424BF9212
        HEX  242444633C373E36363636363636363636362600
CRAYON  HEX  653F2C2D0C6464246F24273F172EDEF6F6363500
DOC     HEX  2D0E2E2D2D3536363E3F3F3F3F3F2724242C45B328
             2D0500
FEUILLE HEX  36363636362D2D2D2D252424243F27240E0EDC183F
             3F0600
```

Source SOURIS Assembleur ProCODE

```
*********************************************************
*                                                       *
*          Programme de gestion de souris               *
*                                                       *
*********************************************************

        ORG  $9000

SET     EQU  $C412
SERVE   EQU  $C413
READ    EQU  $C414
CLEAR   EQU  $C415
POS     EQU  $C416
CLAMP   EQU  $C417
HOME    EQU  $C418
INIT    EQU  $C419

CADRE   EQU  $0
X1      EQU  $6
X2      EQU  $7
Y       EQU  $8
XX1     EQU  $2078
XX2     EQU  $2079
YY1     EQU  $207A
YY2     EQU  $207B

F1      EQU  $2
F2      EQU  F1+1

XDRAW   EQU  $F65D
POSXY   EQU  $F411
SCALE   EQU  $E7

BIT     =    $95F8
RES     =    $95F9
BIT2    =    $95FA
RES2    =    $95FB
```

```
COM     =    $95FC

FIRST   LDA  #1          ;initialisation de la souris
        LDX  #<INIT
        JSR  CALL
        LDA  #$01
        LDX  #<SET
        JSR  CALL        ;en mode passif
        LDX  #<CLEAR
        JSR  CALL        ;remise a zero des coordonnees
        LDA  #0
        STA  X1
        STA  X2
        STA  Y
        RTS              ;fin de l'initialisation

DEBUT   PHA
        JSR  FLECHE1
        PLA
        ASL
        TAY
        LDA  T1,Y
        LDX  T1+1,Y
        STX  T2+1
        STA  T2+2
T2      JSR  COM0
        JMP  LECTURE

T1      DDB  COM0        ;ordre poids fort / poids faible
        DDB  COM1
        DDB  COM2
        DDB  COM3
        DDB  COM4
        DDB  COM5
        DDB  COM6

COM0    LDA  #$80
        STA  BIT
        STA  BIT2
        STA  RES
        STA  RES2
        LDA  #0
        STA  COM
        RTS
COM1    LDA  #$80
        STA  BIT
        STA  BIT2
        LDA  #0
        STA  RES
        STA  RES2
        STA  COM
        RTS
COM2    LDA  #$80
        STA  BIT
        STA  BIT2
        STA  COM
        LDA  #0
        STA  RES
        STA  RES2
        RTS
COM3    LDA  #$80
        STA  BIT
        STA  BIT2
        STA  RES
        STA  RES2
        STA  COM
        RTS
COM4    LDA  #0
        STA  BIT
        STA  BIT2
        STA  RES2
        LDA  #$FF
        STA  RES
        STA  COM
        RTS
COM5    LDA  #$C0
        STA  BIT
        STA  BIT2
        LDA  #$80
        STA  RES
        STA  RES2
        LDA  #0
        STA  COM
        RTS
COM6    LDA  #0
        STA  BIT
        STA  BIT2
        STA  RES
        STA  RES2
        LDA  #$FF
        STA  COM
        RTS
LECTURE LDX  #<READ
        JSR  CALL        ;read mouse
        LDA  $57C
        BEQ  XOK
        CMP  #1
        BEQ  XOK
        CMP  #2
        BNE  XFAUX
        LDA  $47C
        CMP  #$2F
        BCC  XOK
XFAUX   LDA  #2
        STA  $57C
        LDA  #$2F
        STA  $47C
XOK     LDA  $5FC
        BEQ  YOK
        CMP  #1
        BNE  YFAUX
        LDA  $4FC
        CMP  #$7F
        BCC  YOK
YFAUX   LDA  #1
        STA  $5FC
        LDA  #$7F
        STA  $4FC
YOK     LDX  #<POS
        JSR  CALL
        LDA  $47C
        STA  XX1
        LDA  $57C
        STA  XX2
        LDA  $5FC
        STA  YY2
        LDA  $4FC
        STA  YY1
        LSR  XX2
        ROR  XX1
        LSR  YY2
        ROR  YY1
        LDA  COM
        BEQ  A2

        LDY  #2
        LDA  YY1
        CMP  (CADRE),Y
        BEQ  S1
        BCS  S1          ;si superieur a YMIN
        JMP  R2
S1      LDY  #5
        CLC
        CMP  (CADRE),Y
        BEQ  S2
        BCC  S2          ;si inferieur a YMAX
        JMP  R2

S2      LDY  #1          ;maintenant on teste X
        LDA  XX2
        CMP  (CADRE),Y
        BNE  S3
        LDY  #0
        LDA  XX1
        CMP  (CADRE),Y
        BEQ  S4
        BCS  S4
        JMP  R2
S3      LDY  #1
        CMP  (CADRE),Y
        BCS  S4
        JMP  R2
```

```
S4      LDY  #4
        LDA  XX2
        CMP  (CADRE),Y
        BNE  S5
        LDY  #3
        LDA  XX1
        CMP  (CADRE),Y
        BEQ  A2          ;ok
        BCC  A2          ;ok aussi
S5      CMP  (CADRE),Y
        BCC  A2          ;termine
        BEQ  A2
        JMP  R2

A2      LDA  $77C
        AND  BIT
        CMP  RES
        BEQ  RETOUR
A4      LDA  $77C
        AND  #$20
        BEQ  A5
A3      JSR  FLECHE1
        JSR  FLECHE
A5      JMP  LECTURE

RETOUR  LDA  $77C
        AND  BIT2
        CMP  RES2
        BNE  A4
R2      JSR  FLECHE1
        LDA  XX1
        STA  X1
        LDA  XX2
        STA  X2
        LDA  YY1
        STA  Y
        CLC
        RTS
CALL    STX  C400+1
C400    LDX  $C400
        STX  C1+1
        LDX  #$C4
        LDY  #$40
C1      JMP  $C400

FLECHE  LDA  XX1
        STA  X1
        LDA  XX2
        STA  X2
        LDA  YY1
        STA  Y

FLECHE1 LDX  X1
        LDY  X2
        LDA  Y
        JSR  POSXY
        LDA  #1
        STA  SCALE
        LDA  #0
        LDY  F2
        LDX  F1
        JMP  XDRAW
```

Source INVERSION Assembleur ProCODE

```
*********************************
*                               *
* programme d'inversion d'une   *
*                               *
* zone correspondant aux octets *
*                               *
* delimitant la zone.           *
*                               *
*********************************

        ORG  $95C0

**************************************
*                                    *
*  (0),0 : X ( x 8 )                 *
*  (0),1 : Y                         *
*  (0),2 : L ( longueur de la zone ) *
*  (0),3 : H ( hauteur de la zone )  *
*                                    *
**************************************

B0      LDY  #1
        LDA  (0),Y       ;charge Y
        LDX  #0
        LDY  #0
        JSR  $F411
        LDY  #2
        LDA  (0),Y
        TAX
        LDY  #0
        LDA  (0),Y
        TAY
        DEY
B1      INY
        LDA  ($26),Y
        EOR  #$FF
        STA  ($26),Y
        DEX
        BNE  B1
B2      LDY  #1
        LDA  (0),Y
        TAX
        INX
        TXA
        STA  (0),Y
        LDY  #3
        LDA  (0),Y
        TAX
        DEX
        BEQ  FINI
        TXA
        STA  (0),Y
        JMP  B0
FINI    RTS
```

Récapitulation SETCURSOR.C

Après avoir saisi ces codes sous moniteur, vous les sauvegarderez par :
BSAVE SETCURSOR.C, A$94E7, L212

```
94E7:0A
94E8:AA BC F5 94 BD F4 94 85
94F0:03 84 02 60 95 04 95 15
94F8:95 2F 95 44 95 4F 95 77
9500:95 8B 95 A3 2D 2D 1E 3F
9508:37 2D 35 3F 37 4D 0E 0E
9510:25 27 27 06 00 2E 2E 2E
9518:2E 2E 0E AE 65 E4 FF 1B
9520:16 F6 E7 64 6D 40 43 43
9528:2B 26 20 17 24 06 00 2E
9530:2D 0C 3F 27 2D 25 24 24
9538:24 24 24 24 0E 0E 2E 3E
9540:2E 1E 36 00 20 B4 0A 2D
9548:D7 33 26 C0 3F 05 00 49
9550:E4 0D FC 1C 2D 2D 24 24
9558:24 24 24 24 24 24 24 24
9560:24 BF 92 12 24 24 44 63
9568:3C 37 3E 36 36 36 36 36
9570:36 36 36 36 36 26 00 65
9578:3F 2C 2D 0C 64 64 24 6F
9580:24 27 3F 17 2E DE F6 F6
9588:36 35 00 2D 0E 2E 2D 2D
9590:35 36 36 3E 3F 3F 3F 3F
9598:3F 27 24 24 2C 45 B3 28
95A0:2D 05 00 36 36 36 36 36
95A8:2D 2D 2D 2D 25 24 24 24
95B0:3F 27 24 0E 0E DC 18 3F
95B8:3F 06 00
```

Récapitulation SOURIS.C

Après avoir saisi ces codes sous moniteur, vous les sauvegarderez par
BSAVE SOURIS.C, A$9000, L512

```
9000:A9 01 A2 19 20 CB 91 A9
9008:01 A2 12 20 CB 91 A2 15
9010:20 CB 91 A9 00 85 06 85
9018:07 85 08 60 48 20 EA 91
9020:68 0A A8 B9 35 90 BE 36
9028:90 8E 30 90 8D 31 90 20
9030:43 90 4C CF 90 90 43 90
9038:57 90 6B 90 7F 90 91 90
9040:A5 90 BB A9 80 8D F8 95
9048:8D FA 95 8D F9 95 8D FB
9050:95 A9 00 8D FC 95 60 A9
9058:80 8D F8 95 8D FA 95 A9
9060:00 8D F9 95 8D FB 95 8D
9068:FC 95 60 A9 80 8D F8 95
9070:8D FA 95 8D FC 95 A9 00
9078:8D F9 95 8D FB 95 60 A9
9080:80 8D F8 95 8D FA 95 8D
9088:F9 95 8D FB 95 8D FC 95
9090:60 A9 00 8D F8 95 8D FA
9098:95 8D FB 95 A9 FF 8D F9
90A0:95 8D FC 95 60 A9 C0 8D
90A8:F8 95 8D FA 95 A9 80 8D
90B0:F9 95 8D FB 95 A9 00 8D
90B8:FC 95 60 A9 00 8D F8 95
90C0:8D FA 95 8D F9 95 8D FB
90C8:95 A9 FF 8D FC 95 60 A2
90D0:14 20 CB 91 AD 7C 05 F0
90D8:19 C9 01 F0 15 C9 02 D0
90E0:07 AD 7C 04 C9 2F 90 0A
90E8:A9 02 8D 7C 05 A9 2F 8D
90F0:7C 04 AD FC 05 F0 15 C9
90F8:01 D0 07 AD FC 04 C9 7F
9100:90 0A A9 01 8D FC 05 A9
9108:7F 8D FC 04 A2 16 20 CB
9110:91 AD 7C 04 8D 78 20 AD
9118:7C 05 8D 79 20 AD FC 05
9120:8D 7B 20 AD FC 04 8D 7A
9128:20 4E 79 20 6E 78 20 4E
9130:7B 20 6E 7A 20 AD FC 95
9138:F0 57 A0 02 AD 7A 20 D1
9140:00 F0 05 B0 03 4C B7 91
9148:A0 05 18 D1 00 F0 05 90
9150:03 4C B7 91 A0 01 AD 79
9158:20 D1 00 D0 0E A0 00 AD
9160:78 20 D1 00 F0 0E B0 0C
9168:4C B7 91 A0 01 D1 00 B0
9170:03 4C B7 91 A0 04 AD 79
9178:20 D1 00 D0 0B A0 03 AD
9180:78 20 D1 00 F0 0B 90 09
9188:D1 00 90 05 F0 03 4C B7
9190:91 AD 7C 07 2D F8 95 CD
9198:F9 95 F0 10 AD 7C 07 29
91A0:20 F0 06 20 EA 91 20 DB
91A8:91 4C CF 90 AD 7C 07 2D
91B0:FA 95 CD FB 95 D0 E5 20
91B8:EA 91 AD 78 20 85 06 AD
91C0:79 20 85 07 AD 7A 20 85
91C8:08 18 60 8E CF 91 AE 00
91D0:C4 8E D9 91 A2 C4 A0 40
91D8:4C 00 C4 AD 78 20 85 06
91E0:AD 79 20 85 07 AD 7A 20
91E8:85 08 A6 06 A4 07 A5 08
91F0:20 11 F4 A9 01 85 E7 A9
91F8:00 A4 03 A6 02 4C 5D F6
```

Récapitulation INTERFACE.C

Après avoir saisi ces codes sous moniteur, vous les sauvegarderez par :
BSAVE INTERFACE.C, A$9200, L739

```
9200:A9 4C 8D F5 03 A9 10 8D
9208:F6 03 A9 92 8D F7 03 60
9210:C9 B5 F0 11 C9 49 F0 16
9218:C9 43 F0 0F C9 83 F0 08
9220:C9 9E F0 19 60 4C 69 93
9228:4C F2 92 4C A0 92 A0 94
9230:A2 B7 20 8F 94 B0 ED 20
9238:00 90 4C B1 00 20 B1 00
9240:C9 28 D0 E0 20 B1 00 20
9248:67 DD 20 0C E1 A5 A1 8D
9250:DD 94 20 B1 00 20 67 DD
9258:20 0C E1 A5 A1 8D DE 94
9260:20 B7 00 C9 3A D0 BD 20
9268:B1 00 C9 E3 D0 B6 20 B1
9270:00 C9 D0 D0 AF 20 B1 00
9278:20 67 DD 20 0C E1 A5 A1
9280:8D DF 94 20 B1 00 20 67
9288:DD 20 0C E1 A5 A1 8D E0
9290:94 A9 94 85 01 A9 DD 85
9298:00 20 C0 95 4C B1 00 60
92A0:20 B1 00 C9 55 F0 1B A0
92A8:94 A2 C3 20 8F 94 B0 EF
92B0:AD 12 C4 8D BD 92 A9 00
92B8:A2 C4 A0 40 20 00 C4 4C
92C0:B1 00 20 B1 00 C9 52 D0
92C8:D6 20 B1 00 C9 53 D0 CF
92D0:20 B1 00 C9 28 20 B1 00
92D8:20 67 DD 20 0C E1 A5 A0
92E0:D0 0F A5 A1 20 E7 94 20
92E8:B1 00 C9 29 D0 03 20 B1
92F0:00 60 20 B1 00 C9 28 D0
92F8:F8 20 B1 00 20 E3 DF A5
9300:83 8D B5 94 A5 84 8D B6
9308:94 20 B1 00 20 E3 DF A5
9310:83 8D DD 94 A5 84 8D DE
9318:94 20 B1 00 20 E3 DF A5
9320:83 8D DF 94 A5 84 8D E0
9328:94 20 B7 00 C9 29 D0 C1
9330:A9 06 20 1C 90 A4 08 20
9338:01 E3 AE DD 94 AC DE 94
9340:20 2B EB A5 07 A4 06 20
9348:F2 E2 AE B5 94 AC B6 94
9350:20 2B EB AD 7C 07 29 80
9358:A8 20 01 E3 AE DF 94 AC
9360:E0 94 20 2B EB 4C B1 00
9368:60 20 B1 00 C9 28 D0 F8
9370:20 B1 00 A6 B8 8E B5 94
9378:A6 B9 8E B6 94 A0 94 A2
9380:D1 20 8F 94 B0 08 A9 00
9388:20 1C 90 4C B1 00 AE B5
9390:94 86 B8 AE B6 94 86 B9
9398:A0 94 A2 CF 20 8F 94 B0
93A0:08 A9 01 20 1C 90 4C B1
93A8:00 AE B5 94 86 B8 AE B6
93B0:94 86 B9 20 B7 00 C9 BF
93B8:D0 15 20 B1 00 A0 94 A2
93C0:D1 20 8F 94 B0 A2 A9 05
93C8:20 1C 90 4C B1 00 60 AE
93D0:B5 94 86 B8 AE B6 94 86
93D8:B9 A0 94 A2 D8 20 8F 94
93E0:D0 86 20 B1 00 20 67 DD
93E8:20 0C E1 A5 A1 8D DD 94
93F0:A5 A0 8D DE 94 20 B7 00
93F8:C9 2C D0 D2 20 B1 00 20
9400:67 DD 20 0C E1 A5 A1 8D
9408:DF 94 20 B7 00 C9 C1 D0
9410:7D 20 B1 00 20 67 DD 20
9418:0C E1 A5 A1 8D E0 94 A5
9420:A0 8D E1 94 20 B7 00 C9
9428:2C D0 63 20 B1 00 20 67
9430:DD 20 0C E1 A5 A1 8D E2
9438:94 A9 DD 85 00 A9 94 85
9440:01 20 B7 00 C9 5D D0 46
9448:20 B1 00 C9 29 F0 32 C9
9450:CE D0 3B 20 B1 00 A6 B8
9458:8E B5 94 A6 B9 8E B6 94
9460:A2 D1 A0 94 20 8F 94 90
9468:1D AE B5 94 86 B8 AE B6
9470:94 86 B9 A2 CF A0 94 20
9478:8F 94 B0 12 A9 02 4C 88
9480:94 A9 04 4C 88 94 A9 03
9488:20 1C 90 4C B1 00 60 8E
9490:A1 94 8C A2 94 8E A7 94
9498:8C A8 94 A2 00 20 B7 00
94A0:DD 00 02 D0 0E E8 BD 00
94A8:02 F0 06 20 B1 00 4C A0
94B0:94 18 60 38 60 00 00 49
94B8:4E 49 54 28 4D 4F 55 53
94C0:45 29 00 4C 4F 53 45 28
94C8:4D 4F 55 53 45 29 00 4E
94D0:4F 50 52 45 53 53 29 00
94D8:4F 55 54 5B 00 00 00 00
94E0:00 00 00
```

Récapitulation INVERSION.C

Après avoir saisi ces codes sous moniteur, vous les sauvegarderez par :
BSAVE INVERSION.C, A$95C0, L56

```
95C0:A0 01 B1 00 A2 00 A0 00
95C8:20 11 F4 A0 02 B1 00 AA
95D0:A0 00 B1 00 A8 88 C8 B1
95D8:26 49 FF 91 26 CA D0 F6
95E0:A0 01 B1 00 AA E8 8A 91
95E8:00 A0 03 B1 00 AA CA F0
95F0:06 8A 91 00 4C C0 95 60
```

Humeur...

...ou plutôt mauvaise humeur ! Pom's publie suffisamment de sources écrits avec l'assembleur ProCODE pour ne pas être soupçonné d'a priori défavorable... Bien que doté de qualités, bien qu'assemblant sans problème, bien que gérant plusieurs fichiers en mémoire, bien que produit français, ProCODE est ergonomiquement affligeant :

Impossible d'annuler une suppression de ligne involontaire,
Impossible de supprimer directement la dernière ligne,
Impossible d'insérer une ligne avant la première (il faut la dupliquer),
Facile de sauvegarder un fichier en écrasant un ancien portant le même nom (pourtant ProDOS permet de savoir aisément si un fichier existe déjà : pourquoi ne pas prévenir l'utilisateur...),
Faciles les erreurs dans les noms de fichiers : la saisie n'est pas contrôlée,
Facile de tout perdre : on vide la mémoire sans proposer au préalable la sauvegarde,
Impossible d'imprimer sans assembler (agréable en cours de conception lorsque toutes les étiquettes ne sont pas définies...),
Obligé de sortir du programme pour changer le port de l'imprimante,
Impossible de sauvegarder sous le nom /UNIDISK3.5/FICHIER3 car on ne peut dépasser 15 caractères : obligé de définir un préfixe,
Impossible de déverrouiller un fichier : l'option supprimer s'en trouve limitée...

Bientôt une version agréable à utiliser ?

Mathématiques, Physique et Basic,

par Thierry Robin, Éditions Ellipses
156 pages

Après une vingtaine de pages générales sur l'informatique et le Basic, puis une quarantaine de pages sur l'analyse numérique, l'auteur montre comment résoudre, à l'aide de programmes en Basic, 35 problèmes scientifiques, dont des problèmes du concours d'entrée à l'X. C'est dire que la lecture de ce livre, pour intéressante qu'elle soit, risque de rebuter ceux qui n'ont pas de bonnes bases mathématiques. Le titre est d'ailleurs un peu trompeur : à de rares exceptions près, il s'agit de problèmes de mathématiques. Le livre est clair et bien fait ; il aurait toutefois gagné à être présenté avec des listings utilisant des indentations.

Reliures Pom's

Pour la protection de votre précieuse documentation, pour en faciliter la consultation, pour rendre agréable la recherche d'un CALL ou d'un PEEK égaré...

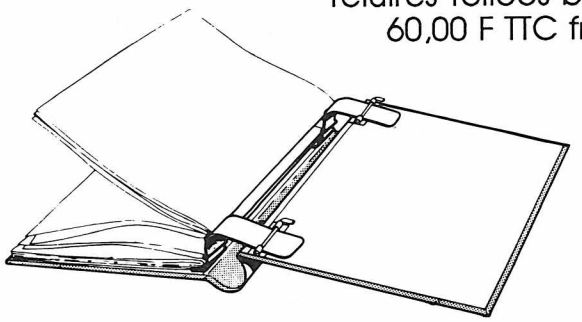
Cryptage... Erratum

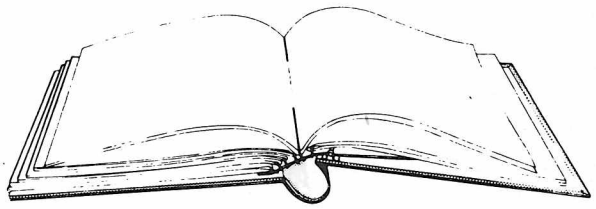
Une erreur s'est glissée dans l'article d'accompagnement de la commande externe 'KRUPTOS' du précédent numéro. Cryptage et décryptage s'obtiennent par la commande :

`CRYPTE nom_fichier, clef [,S n° slot] [,D n° drive]`

On pouvait le déduire par la lecture de la fin du source. Ceci n'a pas empêché trois lecteurs de découvrir la clef de codage du fichier 'SECRET' : «RM.7» ; le bit de poids fort, toujours identique sur les fichiers AppleWriter, était un indice non négligeable, bravo.

reluires toilées bordeaux, logo Pom's, pour 6 numéros : 60,00 F TTC franco, bon de commande page 74





MultiScribe 2.0

Un traitement de textes en DHGR

Bernard Toméno

MultiScribe tourne sur Apple //e 128Ko, //c et IIGS. Il fonctionne sous ProDOS, en double haute résolution graphique, avec ou sans souris et use largement des fenêtres et menus déroulants : bref, un vrai programme Macintosh mais sur Apple //, ce qui est une belle performance.

Bien sûr, les affichages, la rapidité de réaction, le déroulement des menus gagnent en agrément sur le IIGS.

Le programme occupe les deux faces d'une disquette 140Ko qui doit rester en ligne, car le système a recours au disque pour changer de police de caractères, stocker un 'copier', etc.

Pour plus de confort, la souris et deux lecteurs de disquettes sont souhaitables. L'ensemble des fichiers gagne à être placés sur une disquette 800Ko.

L'utilisation est particulièrement intuitive : MultiScribe est une réplique fidèle du traitement de textes MacWrite disponible sur le Macintosh. Les conventions d'ergonomie adoptées sur cette machine ont été fidèlement respectées.

La typographie

Ce qui fait la force et la beauté de MultiScribe : dix polices de caractères en proportionnel dont l'inévitable Gothique. L'utilisateur en créé autant qu'il le souhaite grace à l'éditeur livré sur la même disquette.
Pour chacune d'elles, il est possible de définir 5 tailles de caractères et 10 styles de présentation (voir l'exemple).
Chaque type de police, taille et style peut être choisi avant d'être utilisé ou bien redéfini *a posteriori* sur un texte déjà existant (qui a pu être créé à l'aide d'Applework, Epistole Apple Writer, etc).

Fonctionnalités

On trouve sur MultiScribe les classiques fonctions de définition de marges (droite et gauche), justification (droite, gauche, totale, centrée), espacement de lignes variable, chercher et remplacer, couper/copier/coller.
Il faut aussi noter les possibilités de définir des hauts et bas de pages automatiques (pouvant inclure date et heure si vous avez la carte horloge), une numérotation automatique des pages, l'insertion de saut de pages...
Le déplacement dans le texte s'obtient par la touche ⌘> ou ⌘<, ⌘chiffre (comme dans Applework) ou à l'aide de la souris et du classique ascenseur latéral.

Environnement

On peut choisir son interface (45 propositions, séries ou parallèles) et son imprimante (11 machines dont ImageWriter, DMP, Epson FX, RX et MX, Okidata 92IBM, 92STD, Scribe, Brother) et sauvegarder ce choix sur disque.
On peut aussi définir son format de papier (4 possibilités, la plus utile en France étant Fanfold International).
Les textes peuvent être sauvegardés sur tous supports (merci ProDOS), sur 5 1/4 ou 3 1/2, disque dur, ou disque virtuel volatile, ceci en format MultiScribe ou en format TEXT (ASCII). Il existe des options pour formater une disquette et détruire un fichier.

L'impression

Cinq qualités d'impression papier sont offertes, de plus en plus noires et lisses, le temps d'impression croissant avec la qualité, plus une qualité courante non graphique, plus une visualisation à l'écran avant d'imprimer. Pour la meilleure qualité, la tête d'impression passe quatre fois sur la même ligne.
Il y a la possibilité de choisir le feuille à feuille ou le papier continu, l'impression totale ou partielle d'un texte, l'impression de plusieurs documents chaînés, la numérotation automatique des pages.

L'éditeur de polices

D'une fort belle présentation, il permet de créer ses propres polices, ou de modifier celles qui sont sur la disquette (francisation facile de certains caractères typiquement américains, création de lettres accentuées, de symboles scientifiques, ou même de petits logo ou dessins divers comme ☞ par exemple).

Options

Pour quelques dollars de plus vous pouvez acheter des "options" :

- Accessoires de bureau type Macintosh (horloge, calendrier, calculette...) ;
- Jeux de polices de caractères supplémentaires ;
- 'Picture Manager' pour mélanger dessins et texte ;
- Correcteur d'orthographe (in english, sorry...).

Points forts, regrets

Au chapitre points forts, on trouve la qualité de l'ergonomie et de la présentation à l'écran graphique, les nombreuses possibilités de typographie et d'impression, le couper/coller, l'ouverture des fichiers par une fenêtre de sélection : nul besoin de se rappeler l'orthographe exacte des nom de fichiers.
Les menus déroulants dispensent l'utilisateur de se rappeler la syntaxe de telle ou telle fonction : le menu et la fenêtre de remplacement sont plus plaisants que :

```
CTRL-F %$paris%$Paris%a...
```

Rien n'étant parfait (quoi que...) en ce bas monde, on peut regretter :

- l'absence de glossaire ;
- les possibilités très limitées de mélanger plusieurs textes ou fragments de texte à l'écran (on annonce une version IIGS qui le pourrait) ;
- l'impossibilité de sauvegarder sur disque les informations concernant le format du papier ;
- l'impossibilité d'annuler la dernière commande ;
- l'absence d'un WPL, ce langage d'AppleWriter qui permet d'automatiser les commandes répétitives, faire des mailings, etc.

Dernier problème : sur un Apple //e avec Carte Féline, MultiScribe ne fonctionne pas (illisibilité à cause d'un problème d'affichage en double haute résolution). Don't panic... la solution est à la fin de ce papier.

Commercialisation

MultiScribe est vendu par Styleware inc. (5250 Gulfton suite E, Houston, Texas 77081) à un prix défiant toute concurrence (60 dollars). Il suffit d'écrire en précisant son numéro de carte de crédit et la date d'expiration : 15 jours après, MultiScribe est chez vous avec un manuel clair et bien présenté.
La disquette n'est pas protégée comme bien des softs américains aujourd'hui : mais MultiScribe est si bien fait et si bon marché qu'on ne doit pas avoir envie de le copier pour les copains. Ceux qui l'on écrit méritent un coup de chapeau et notre honnêteté.

Recette pour un Apple //e avec carte Féline

Avec la disquette ARLEQUIN, créez une disquette SYSTEM (elle contiendra donc PRODOS, ARLEQUIN.SYSTEM qui est un Basic amélioré, ainsi que les fichiers GLIDATA, GLI16.2, GLI.ERRORS, IWIICOLOR-COPY) et donnez lui le nom de /MULTISCRIBE.
Sauvez y un STARTUP ainsi conçu :

```
10 D$ = CHR$(4) : PRINT D$
   "PREFIX/MULTISCRIBE"
20 & BACK (17) : & WINDOW
   ( 0,19,0,191,1 ) : & MO
   DE (1) : & SETVAR ( 50,
   16 ) :& DISPLAY (9) : P
   OKE 49234,0 : PRINT D$
   "BLOAD MS.SYSTEM, TSYS,
    A$2000 "
30 & DISPLAY ( 1 ) : CALL
   8192
```

Puis recopier sur la disquette les fichiers de la face 1 du disque MULTISCRIBE, et au verso ceux de la face 2. C'est tout.

MULTISCRIBE 2.0 Quelques possibilités d'un traitement de textes en DHGR

Dix polices de caractères mixables à l'intérieur d'un même texte sans perdre la justification totale, y compris avec le ê. Dix polices de caractères mixables à l'intérieur **d'un même texte sans perdre la justification totale, y compris avec** le ê. Dix polices de caractères mixables à l'intérieur d'un même **texte sans perdre la justification totale, y compris avec le ê. Dix polices de caractères mixables à l'intérieur d'un même texte sans perdre la justification totale, y compris avec le ê.**

Un caractère peut être standard, gras, *italique*, souligné, ombré, relief, inversé, indice, exposant, double hauteur, double largeur. Ces différents arguments sont compatibles entre-eux : *gras italique souligné*, relief double largeur...

DhgrTool : un outil graphique

Roland Jost

La double haute résolution graphique permet de disposer de 560 x 192 points sur un Apple //c ou un //e avec mémoire étendue à 128Ko. Mais l'Applesoft ne présente aucune facilité pour son accès et son utilisation reste délicate.

Une image DHGR réside entre $2000 et $3FFF en mémoire principale et entre $2000 et 3FFF en mémoire auxiliaire, occupant 16Ko de MEV. Toutes les colonnes paires de l'écran 80 colonnes correspondent à la page auxiliaire, toutes les colonnes impaires à la page principale. Le microprocesseur de l'Apple ne pouvant adresser que 64Ko, l'accès à la mémoire auxiliaire ne peut se faire que par l'intermédiaire de bascules logicielles.

Chargement et sauvegarde d'images DHGR

Il est possible de stocker une image DHGR dans deux fichiers de 8Ko chacun, correspondant à la page graphique 1 en mémoire principale et la page graphique 1 en mémoire auxiliaire (Cf les images de Purplesoft).

Dans ce cas, le chargement se fait en deux fois : une image est chargée en mémoire principale, puis transférée dans la page 1 auxiliaire en utilisant la routine AUXMOVE. Le deuxième fichier est alors chargé en page 1 de la mémoire principale.

DhgrTool simplifie le processus en chargeant directement l'image en mémoire auxiliaire à partir du disque, sans passage par la mémoire principale.

Il paraît plus logique de sauver l'image DHGR dans un fichier unique. Pour cela, DhgrTool transfère la page 1 auxiliaire en page 2 principale et l'image est sauvegardée par copie de l'ensemble des 2 pages HGR principales sur le disque (`BSAVE IMAGE, A$2000, L$3FF8`).

Le fichier comporte 33 blocs sous ProDOS ou 67 secteurs sous DOS 3.3. Le processus inverse a lieu au chargement : la page HGR2 principale est transférée dans la page 1 auxiliaire.

Tous les vétérans de l'Apple disposent d'images graphiques simple haute résolution. Celles-ci peuvent être converties de deux façons :

- deux images HGR combinées en une image DHGR : on charge successivement chacune des images HGR en page 2 et le sous-programme `HGHDHGR` transfère chacune des 40 colonnes alternativement en page 1 auxiliaire et 1 principale.

`CALL H1,0` transfère la première image

`CALL H1,20` la deuxième

- conversion d'une image HGR en image DHGR : le sous-programme `DBLHGR` dupli-que horizontalement chaque point de l'image HGR. Chaque octet de l'image initiale donne deux octets dont l'un est chargé en 1 auxiliaire et l'autre en page 1 principale.

Manipulation d'images DHGR par DhgrTool

1—afficher une image

Les opérations suivantes sont nécessaires :

- déconnecter AN3 par lecture de l'adresse 49246
- déconnecter ALPZP par POKE 49160,0, qui permet de commuter uniquement entre les adresses $2000-$BFFF
- passer en mode 80 colonnes par POKE 49165,0 (80 COL ON)
- déconnecter TEXT en *peekant* en 49232
- *peeker* en 49239 (HIRES ON) pour activer le graphisme haute résolution
- *peeker* en 49234 (MIXED OFF) pour avoir accès à la totalité de l'écran.

2—effacer l'image

- effacer la page 1 par le HGR classique
- commuter la page 1A (auxiliaire) par POKE 49237,0 (PAGE2 ON)
- effacer la page 1A par CALL 62450
- revenir en page 1 par POKE 49236,0 (PAGE2 OFF).

3—négatif de l'image

Le programme assembleur effectue un `EOR $FF` sur tous les octets de la page HGR 1 principale et de la page 1 auxiliaire

4—sauvegarde en mémoire auxiliaire et rappel

Une image peut être sauvée dans la mémoire auxiliaire en $4000-7FFF ou $8000-BFFF. Le sous-programme `TRANSFERT` permet les déplacements suivants :

Pour le transfert de 2 pages

`CALL TRANSFERT,2,2`
2A vers 1 ; 3A vers 1A

`CALL TRANSFERT,3,2`
1 vers 2A ; 1A vers 3A

`CALL TRANSFERT,4,2`
4A vers 1 ; 5A vers 1A

`CALL TRANSFERT,5,2`
1 vers 4A ; 1A vers 5A

Pour le transfert d'une seule page

`CALL TRANSFERT,0,1`
1A vers 2

`CALL TRANSFERT,1,0`
2 vers 1A

Pour les deux derniers cas, l'appel à TRANSFERT doit être précédé d'un `POKE 49152,0` (80 STORE OFF) et suivi d'un `POKE 49153,0` (80STORE ON). En effet 80STORE ON a priorité sur les commutateurs RAMRD et RAMWRT utilisés par AUX-MOVE pour réaliser le transfert.

D'autres possibilités peuvent être obtenues en complétant ou modifiant les tables DEBUT1, FIN1, NOUV1, DEBUT2, FIN2, NOUV2 (lignes 259 à 264 dans le source)

5—négatif d'une moitié d'image

On peut inverser la moitié gauche ou la moitié droite. DHGRTOOL ne propose que l'inversion de la partie gauche. Pour inverser la partie droite faire d'abord un négatif de toute l'image, puis inverser la partie gauche.

6—échange des moitiés gauche et droite

Intéressant lorsque l'image DHGR a été obtenue à partir de deux images HGR normales, ou pour des effets spéciaux.

7—copie d'une moitié dans l'autre

8—effet de miroir horizontal)

DHGRTOOL.0

Toutes ces opérations sont effectuées par le programme Assembleur DHGRTOOL.0 qui est basé sur l'utilisation de la routine `AUXMOVE` et sur les commutateurs `PAGE2 ON` et `PAGE2 OFF` (qui permettent de travailler respectivement sur la page 1 ou la page 1A).

La routine AUXMOVE

AUXMOVE permet le transfert de blocs de données entre la mémoire principale et la mémoire auxiliaire, dans les deux sens. Elle est installée dans l'espace mémoire $C300-$C3FF et commence en $C311.

Comme la routine MOVE du moniteur , elle utilise les adresses $3C - $43 pour délimiter la zone à transférer et l'adresse d'arrivée.

$3C A1L octet bas de l'adresse de début du bloc à transférer
$3D A1H octet haut de l'adresse de début
$3E A2L octet bas de l'adresse de fin de la zone à transférer
$3F A2H octet haut de l'adresse de fin
$42 A4L octet bas de l'adresse d'arrivée
$43 A4H octet haut de l'adresse d'arrivée

De plus , c'est la valeur de la retenue qui détermine le sens du transfert :

C=1 : mémoire principale vers mémoire auxiliaire
C=0 : mémoire auxiliaire vers mémoire principale

Le commutateur PAGE2

En mode HGR normal, PAGE2 permet l'accès à la page HGR 2. En mode 80 colonnes et double haute résolution, et lorsque 80STORE est actif, un `POKE 49237,0` permet au processeur d'adresser la page 1A , alors qu'un `POKE 49236,0` permet l'adressage de la page 1.

Implantation des divers programmes

$6022 Négatif de la page DHGR
$6047 Renverse l'image DHGR (miroir)
$6099 HGR→DHGR : dilate une image HGR
$60E3 deux images HGR placées côte à côte utilise le sous-programme CALCADR et les tables TAB1L et TAB1H
$6186 échange les deux moitiés de l'image DHGR
$61D2 copie la moitié gauche dans la partie droite
$6216 copie la moitié droite dans la moitié gauche
$625C négatif de la partie gauche
$626F négatif de la partie droite
$62AE transfert de pages graphiques
$633E correction d'erreurs.

HardCopy de l'écran DHGR

Une copie d'écran peut être obtenue avec le programme HARD de Marc Debuigne (Pom's n° 18). Mais la page 2 étant utilisée, il faut le reloger . Il a été choisi de le charger en $1000. Il faut donc *poker* $10 dans les octets $10D0, $10EE, $10F3, $115B, $1165, $1187.

En basic : `POKE 4304,16: POKE 4334,16: POKE 4339,16: POKE 4443,16: POKE 4453,16: POKE 4487,16.`

Le programme HARD original permet d'imprimer une image en deux tailles 560*192 (1:1) et 1120*364 (2:2). En masquant quelques instructions, on dispose de deux possibilités supplémentaires : (2:1) et (1:2). Voici la méthode :

Pour image 2:1

Faire les pokes suivants : `POKE 768,128: FOR I=0 TO 4: POKE 4423+I,234: NEXT.` Ceci revient à mettre NOP dans les adresses $1141 à $1145.

Pour image 1:2

`POKE 768,128: FOR I=5 TO 7 : POKE 4423+I,234: NEXT.` On met NOP dans les adresses $1146 et $1147.

Après la hardcopy il faut restaurer le programme initial ($1141: 2C 00 03 10 03 20 ED FD) par la séquence : `POKE 4423,44: POKE 4424,0: POKE 4425,3: POKE 4426,16: POKE 4427,3: POKE 4428,32: POKE 4429,237: POKE 4430,253.`

Le programme HARD original présente un inconvénient lorsqu'il est utilisé sous ProDOS, car il déconnecte ce dernier. En effet, HARD modifie les pointeurs CSWL/CSWH en page zéro , ce qui est à éviter. Il est préférable

de passer par les adresses $BE30 du BI (ProDOS) (cf. Beneath Apple ProDOS). Les pointeurs sont modifiés par les routines ACTIVE et DESACTIV de DHGRTOOL0. HARD est *patché* en conséquence (voir la ligne 20097 dans DHGRTOOL).

Utilisation de DHGRTOOL.O dans d'autres programmes

DHGRTOOL.0 est implanté à partir de $800, à la place habituelle d'un programme BASIC. Ce dernier devra donc être placé après la page HGR 2, en $6000. Prendre exemple sur la ligne 10 de DHGRTOOL.

Vous pouvez aussi réassembler DHGRTOOL.0 à partir du programme source. En tout cas, vous devez laisser les pages HGR 1 et 2 libres.

Pour disposer de la copie d'écran graphique, le programme HARD (Pom's 18) doit être présent sur la disquette. De plus, un certain nombre de *patches* sont nécessaires pour le reloger et le modifier. Prendre exemple sur les lignes 5070, 5080, 5185, 20095, 20096, 20097.

Certains sous-programmes de DHGRTOOL peuvent être utilisés dans vos propres réalisations :

—gestion des menus ;
—sélection d'une page graphique du disque (lignes 100-290) ;
—routine de saisie de nom de fichier (lignes 300-490) ;
—hardcopy (lignes 5000-5300).

NB : DHGRTOOL ne tourne que sous ProDOS dont il utilise la possibilité de lire le directory d'une disquette et les appels au Basic Interpreter. Sous DOS 3.3, il faudrait modifier la routine de sélection d'une image (ligne 100) et les *patches* au programme HARD concernant les entrées/sorties.
DHGRTOOL.0 est utilisable sous DOS 3.3 à l'exclusion de ACTIVE et DESACTIV.

Source DHGR.S
Assembleur Edasm ProDOS ou ProCODE

(Pour Edasm, LSR, ROR... ne concernent pas implicitement A ; il faut donc les remplacer par LSR A, ROR A)

```
* UTILITAIRES DOUBLE HAUTE RESOLUTION
*
*         R.JOST - 1986
*
* Edasm ProDOS/ProCODE
*
        ORG  $800
*
        JMP  TRANSFERT
        JMP  INVDHGR
        JMP  MIRHOR
        JMP  DBLHGR
        JMP  HGRDHGR
        JMP  ECHANGE
        JMP  COPIEGD
        JMP  COPIEDG
        JMP  INVGAUC
        JMP  INVDROI
        JMP  ERREUR
        JMP  INKEY
        JMP  ACTIVE
        JMP  DESACTIV
                        ;
                        ;
A1L     EQU  $3C        ;
A1H     EQU  $3D
A2L     EQU  $3E
A2H     EQU  $3F
A4L     EQU  $42
A4H     EQU  $43
CHOIX   EQU  $06
                        ;
AUXMOVE EQU  $C311      ; routine de transfert
*
*
* NEGATIF DE LA PAGE DHGR
*
ADRESSE EQU  $06
```

```
IMAGES DOUBLE HAUTE RESOLUTION      -     R.Jost (1986)
 ____________________
!  Menu Principal          !____________________________
!  ____________________                                 !
! !  Chargement d'une image !____________________________
! !                                                      !
! !                                                      !
! !                  Charger une image 16K               !
! !                                                      !
! !                  Charger deux images                 !
! !                                                      !
! !                  Charger deux images HGR             !
! !                                                      !
! !                  Charger une image HGR               !
! !                                                      !
! !                  Retour au MENU principal            !
!_!                                                      !
  !                                                      !
  !______________________________________________________!

Choisir avec les fleches / Valider par <ENTER> / Quitter par <ESC>
```

DHGRTOOL : Présentation type 'AppleWorks' et ergonomie plaisante

ProDOS
][+
//e
//e+
//c
][gs

```
SAUVE    EQU  $09
LINNUM   EQU  $50
MEMPRINC EQU  $C054      ; sélectionne la page 1
MEMAUXI  EQU  $C055      ; sélectionne la page 1A
*
INVDHGR  NOP
*
         LDX  #$00
         STX  ADRESSE
         LDX  #$20
         STX  ADRESSE+1  ; on démarre en $2000
INV0     LDY  #$00
INV1     STA  MEMAUXI
         LDA  (ADRESSE),Y
         EOR  #$FF       ; inverse un octet en 1A
         STA  (ADRESSE),Y
         STA  MEMPRINC
         LDA  (ADRESSE),Y
         EOR  #$FF       ; inverse un octet en 1
         STA  (ADRESSE),Y
         DEY
         BNE  INV1       ; ceci sur 256 octets
```

```
        INC  ADRESSE+1
        DEX
        BNE  INV0       ; et 32 fois.
        RTS
*
* MIROIR HORIZONTAL
*
MIRHOR  NOP
*
Z1      EQU  $06
Z2      EQU  $07
DEPART  EQU  $18
ARRIVEE EQU  $1A
TAMPON  EQU  $1C
X1      EQU  $EB
X2      EQU  $EC
Y1      EQU  $ED
Y2      EQU  $EE
*
*
        LDX  #$00
        STX  X1         ; initialisation des pointeurs
        STX  Y1
        STX  Z1
        LDX  #191       ; 192 lignes
        STX  Z2
        STX  Y2
        LDX  #40        ; 40 octets par ligne
        STX  X2
HOR1    LDX  Z1         ; calcul des adresses de départ
        JSR  CALCADR
        STA  DEPART
        STY  DEPART+1
        LDX  Z2
        JSR  CALCADR    ; et d'arrivée
        STA  ARRIVEE
        STY  ARRIVEE+1
        LDY  X1         ; on traite 1 ligne
HOR2    STA  MEMAUXI    ; on s'occupe de la page 1A
        LDA  (DEPART),Y
        PHA
        LDA  (ARRIVEE),Y
        STA  (DEPART),Y
        PLA
        STA  (ARRIVEE),Y
        STA  MEMPRINC   ; et de la page 1
        LDA  (DEPART),Y
        PHA
        LDA  (ARRIVEE),Y
        STA  (DEPART),Y
        PLA
        STA  (ARRIVEE),Y
        INY
        CPY  X2
        BNE  HOR2       ; fin de la ligne ?
        INC  Z1         ; on passe a la ligne suivante
        DEC  Z2
        SEC
        LDA  Z2
        SBC  Z1
        BCS  HOR1
        RTS
*
* HGR --> DHGR
*
* transforme une image HGR en image DHGR
*
DBLHGR  NOP
*
*
        LDX  #$00
        STX  ADRESSE
        LDX  #$20
        STX  ADRESSE+1  ; l'image est en page 1
BOUC0   LDY  #$00
BOUC1   LDA  (ADRESSE),Y ; on traite tous les octets
        STX  SAUVE
        JSR  DEDOUBLE   ; en dupliquant chaque bit
        LDX  SAUVE
        STA  MEMAUXI    ; le 1er octet est charge en page
1A
        LDA  LINNUM
        STA  (ADRESSE),Y
        STA  MEMPRINC   ; le 2ème en page 1
        LDA  LINNUM+1
        STA  (ADRESSE),Y
        DEY
        BNE  BOUC1
        INC  ADRESSE+1
        DEX
        BNE  BOUC0
        RTS
*
* dédouble chacun des 7 premiers bits d'un octet HGR
* et stocke dans deux octets en conservant le bit de
* couleur. cf. Pascal Cantot, Pom's 20
*
DEDOUBLE LDX  #$00
        STX  LINNUM
        STX  LINNUM+1
        LDX  #$7
DECAL   ASL
        PHP
        JSR  ROLINNUM
        PLP
        JSR  ROLINNUM
        DEX
        BPL  DECAL
        JSR  ROLINNUM
        LSR  LINNUM
        RTS
ROLINNUM ROL  LINNUM
        ROL  LINNUM+1
        RTS
*
* 2 IMAGES HGR
*
* stocke 2 images HGR côte à côte en DHGR
*
* copie l'image  chargée en page 2
* ($4000-$5FFF) dans une moitié de page DHGR
*
* appel par CALL AD,X
*   X = numéro de la 1ère colonne ( 0<=X<=20 )
*
HGRDHGR NOP
*
XA0     EQU  $1C
XA1     EQU  $1D
*
GETBYTC EQU  $E6F5
*
        LDY  #00
        STY  X1
        STY  Y1
        LDY  #40        ; 40 octets
        STY  X2
        LDY  #191       ; 192 lignes
        STY  Y2
        JSR  GETBYTC    ;
        STX  XA0        ; numéro 1ère colonne
        STX  XA1
*
BOUCLE  LDX  Y2
        JSR  CALCADR
        STA  DEPART
        TYA
        CLC
        ADC  #$20
        STA  DEPART+1   ; adresse ligne départ en page 2
        JSR  CALCADR
        STA  ARRIVEE
        STY  ARRIVEE+1  ; adresse ligne en page 1 ou 1A
LIGNE   LDY  X1
        LDA  (DEPART),Y ; 40 octet par ligne
        PHA
        INY
        LDA  (DEPART),Y
```

```
        INY
        STY  X1
        LDY  XA1
        STA  MEMPRINC   ; stocke en page 1
        STA  (ARRIVEE),Y
        PLA
        STA  MEMAUXI    ; stocke en page 1A
        STA  (ARRIVEE),Y
        INC  XA1
        STA  MEMPRINC
        LDA  X1
        CMP  X2
        BNE  LIGNE
        LDA  #$00
        STA  X1
        LDA  XA0
        STA  XA1
        DEC  Y2
        LDA  Y2
        CMP  #$FF
        BNE  BOUCLE
        RTS
*
* calcule l'adresse d'un octet en page HGR à partir
* des tables TAB1L et TAB1H.
*
CALCADR TXA
        LSR
        LSR
        LSR
        TAY
        LDA  TAB1L,Y
        PHA
        TXA
        AND  #$07
        ASL
        ASL
        CLC
        ADC  TAB1H,Y
        TAY
        PLA
        RTS
*
*
TAB1L   DFB  $00,$80,$00,$80,$00,$80,$00,$80
        DFB  $28,$A8,$28,$A8,$28,$A8,$28,$A8
        DFB  $50,$D0,$50,$D0,$50,$D0,$50,$D0
TAB1H   DFB  $20,$20,$21,$21,$22,$22,$23,$23
        DFB  $20,$20,$21,$21,$22,$22,$23,$23
        DFB  $20,$20,$21,$21,$22,$22,$23,$23
*
*
* ECHANGE DE DEUX MOITIES DHGR
*
* copie la moitié gauche dans la droite et vice-versa
*
ECHANGE LDX  #$00
        STX  X1
        STX  Y1
        LDX  #192
        STX  Y2
        LDX  #20
        STX  X2
SWAP1   LDX  Y1
        JSR  CALCADR
        STA  DEPART
        STY  DEPART+1
        LDY  X1
SWAP2   STA  MEMAUXI
        JSR  SWAP3
        STA  MEMPRINC
        JSR  SWAP3
        INY
        CPY  X2
        BNE  SWAP2
        INC  Y1
        LDA  Y1
        CMP  Y2
        BNE  SWAP1
        RTS
*
* échange le contenu de 2 octets DHGR
*
SWAP3   LDA  (DEPART),Y
        STA  TAMPON
        TYA
        PHA
        CLC
        ADC  #20
        TAY
        LDA  (DEPART),Y
        STA  TAMPON+1
        LDA  TAMPON
        STA  (DEPART),Y
        PLA
        TAY
        LDA  TAMPON+1
        STA  (DEPART),Y
        RTS
*
*
* COPIE MOITIE PAGE DHGR GAUCHE DANS DROITE
*
*
COPIEGD LDX  #$00
        STX  X1
        STX  Y1
        LDX  #192
        STX  Y2
        LDX  #20
        STX  X2
COP1    LDX  Y1
        JSR  CALCADR
        STA  DEPART
        STY  DEPART+1
        LDY  X1
COP2    STA  MEMAUXI
        JSR  COP3
        STA  MEMPRINC
        JSR  COP3
        INY
        CPY  X2
        BNE  COP2
        INC  Y1
        LDA  Y1
        CMP  Y2
        BNE  COP1
        RTS

COP3    LDA  (DEPART),Y ; copie l'octet X en 20 + X
        STA  TAMPON
        TYA
        PHA
        CLC
        ADC  #20
        TAY
        LDA  TAMPON
        STA  (DEPART),Y
        PLA
        TAY
        RTS
*
*
* COPIE MOITIE PAGE DHGR DROITE DANS GAUCHE
*
*
COPIEDG LDX  #20
        STX  X1
        LDX  #$00
        STX  Y1
        LDX  #192
        STX  Y2
        LDX  #40
        STX  X2
COPY1   LDX  Y1
        JSR  CALCADR
        STA  DEPART
        STY  DEPART+1
```

```
        LDY  X1
COPY2   STA  MEMAUXI
        JSR  COPY3
        STA  MEMPRINC
        JSR  COPY3
        INY
        CPY  X2
        BNE  COPY2
        INC  Y1
        LDA  Y1
        CMP  Y2
        BNE  COPY1
        RTS
*
COPY3   LDA  (DEPART),Y
        STA  TAMPON
        TYA
        PHA
        SEC
        SBC  #20
        TAY
        LDA  TAMPON
        STA  (DEPART),Y
        PLA
        TAY
        RTS
*
*
* INVERSE MOITIE GAUCHE
*
*
INVGAUC LDX  #00
        STX  X1
        LDX  #$00
        STX  Y1
        LDX  #192
        STX  Y2
        LDX  #20
        STX  X2
        JMP  INV
*
* INVERSE MOITIE DROITE
*
INVDROI LDX  #20
        STX  X1
        LDX  #00
        STX  Y1
        LDX  #192
        STX  Y2
        LDX  #40
        STX  X2
        JMP  INV
*
INV     LDX  Y1
        JSR  CALCADR
        STA  DEPART
        STY  DEPART+1
        LDY  X1
INV2    STA  MEMAUXI
        JSR  INV3
        STA  MEMPRINC
        JSR  INV3
        INY
        CPY  X2
        BNE  INV2
        INC  Y1
        LDA  Y1
        CMP  Y2
        BNE  INV
        RTS
*
INV3    LDA  (DEPART),Y ; inverse un octet
        EOR  #$FF
        STA  (DEPART),Y
        RTS
*
* TRANSFERT DE PAGE DHGR
*
 TRANSFERT NOP
        JSR  GETBYTC
        STX  CHOIX
        JSR  GETBYTC
        CPX  #0
        BNE  TRF1
        JSR  TRANS0
        RTS
TRF1    CPX  #1
        BNE  TRF2
        JSR  TRANS1
        RTS
TRF2    CPX  #2
        BNE  TRFFIN
        JSR  TRANS0
        JSR  TRANS1
TRFFIN  RTS
TRANS0  LDX  CHOIX
        STA  MEMPRINC
        JSR  INITMOVE
        LDA  DEBUT1,X
        STA  A1H
        LDA  FIN1,X
        STA  A2H
        LDA  NOUV1,X
        STA  A4H
        TXA
        ROR
        JSR  AUXMOVE
        RTS
TRANS1  LDX  CHOIX
        JSR  INITMOVE
        LDA  DEBUT2,X
        STA  A1H
        LDA  FIN2,X
        STA  A2H
        LDA  NOUV2,X
        STA  A4H
        STA  MEMAUXI
        TXA
        ROR
        JSR  AUXMOVE
        STA  MEMPRINC
        RTS
                        ;
INITMOVE LDA  #$00      ;Initialise les octets bas
        STA  A1L        ; des adresses utilisées
        STA  A4L        ; par AUXMOVE.
        LDA  #$FF
        STA  A2L
        RTS
                        ;
                        ; tables des adresses hautes
                        ; utilisees par AUXMOVE
DEBUT1  DFB  $20,$40,$40,$20,$80,$20
FIN1    DFB  $3F,$5F,$5F,$3F,$9F,$3F
NOUV1   DFB  $40,$20,$20,$40,$20,$80
DEBUT2  DFB  $20,$20,$60,$20,$A0,$20
FIN2    DFB  $3F,$3F,$7F,$3F,$BF,$3F
NOUV2   DFB  $40,$40,$20,$60,$20,$A0
ERREUR  PLA
        TAY
        PLA
        LDX  $DF
        TXS
        PHA
        TYA
        PHA
        RTS
*
* DRIVER IMPRIMANTE
*
OLDVEC  EQU  $2FE
*
*  Sauve les valeurs dans VECTIN ($BE30-31)
* et les remplace par le port imprimante.
*
*
ACTIVE  LDA  $BE30
        STA  OLDVEC
```

```
        LDA  $BE31
        STA  OLDVEC+1
        LDA  #$00
        STA  $BE30
        LDA  #$C1
        STA  $BE31
        RTS
*
* Redirige les sorties vers la
* carte 80 colonnes (slot 3).
*
DESACTIV LDA  OLDVEC
        STA  $BE30
        LDA  OLDVEC+1
        STA  $BE31
        RTS
*
* SAISIE D'UNE TOUCHE
*
*
*
VARPNT  EQU  $83        ; adresse de la dernière variable
KBD     EQU  $C000
STROBE  EQU  $C010
CHKCOM  EQU  $DEBE
PTRGET  EQU  $DFE3
SGNFLT  EQU  $E301      ; rend flottant l'entier sans signe
MOVMF   EQU  $EB2B      ; transfère FAC dans octet pointé
        par X et Y
*
*
INKEY   JSR  CHKCOM
        JSR  PTRGET
KEY     LDA  KBD
        BPL  KEY
        BIT  STROBE
        CMP  #$FF
        BEQ  KEY1
        CMP  #$E0       ; est-ce une minuscule ?
        BCC  KEY1
        AND  #$DF       ; convertit en majuscule
KEY1    AND  #$7F       ; bit 7 à zéro
        TAY
        JSR  SGNFLT     ; convertit Y en flottant dans FAC
        LDX  VARPNT     ; adresse de la variable
        LDY  VARPNT+1
        JSR  MOVMF      ; transfère FAC dans la variable
        RTS
```

Source HARD.S

Il s'agit du source publié dans le numéro 18 de Pom's.

Programme DHGRTOOL

```
10  IF  PEEK (103) +  PEEK (104) * 256 <  > 2457
     7 THEN  POKE 103,1: POKE 104,96: POKE 2457
     6,0: PRINT  CHR$ (4)"RUN DHGRTOOL"
20  GOTO 20000
100  REM   CHARGEMENT D'UNE IMAGE
110 ER = 0:CX = CX(NM):CY = CY(NM): GOSUB 500: P
     OKE 32,CX + 1: VTAB CY + 3: POKE 33,CX + 5
     0: POKE 34,CY + 2: POKE 35,CY + 15
115  PRINT D$"PREFIX,D1"
120  PRINT D$"PREFIX": INPUT DIR$
130  PRINT D$"OPEN"DIR$",TDIR"
140  PRINT D$"READ"DIR$
145  INPUT T$: INPUT T$: INPUT T$
150  ONERR  GOTO 180
155 I = 0
160  INPUT NAME$:NAME$ = NAME$ + ""
162  IF NAME$ = "" THEN 180
164  IF  MID$ (NAME$,18,3) <  > "BIN" THEN 160
166  IF  MID$ (NAME$,26,3) <  > TA$ THEN 160
168  IF  MID$ (NAME$,76,4) <  > "2000" AND  MID$
      (NAME$,76,4) <  > "4000" THEN 160
172 I = I + 1:S$(4,I) =  LEFT$ (NAME$,15): GOTO
     160
180  POKE 216,0: PRINT D$"CLOSE": IF I > 12 THEN
      I = 12
181 S1 = I: IF S1 = 0 THEN  PRINT  CHR$ (7)" Auc
     une image sur ce disque ";: GET R$: POP :
     RETURN
182 IM = 4:HL = 6:HT = 10
184  GOSUB 1020: IF K = 27 THEN  POP : RETURN
186 NAME$ = S$(4,C1)
190  IF C0 = 2 THEN  RETURN
230  REM
260  PRINT : PRINT D$"BLOAD"NAME$;A$
290  RETURN
300  REM  SAISIE D'UN NOM DE FICHIER
310 :
320  HTAB 1: VTAB 23: TEXT : CALL  - 868: PRINT
     "Tapez le nom de l'image, puis <Enter> //
      <ESC> pour annuler";
330 ER = 0:CX = CX(NM):CY = CY(NM): GOSUB 500: P
     OKE 32,CX + 1: VTAB CY + 3
340  POKE 33,CX + 50
350  POKE 34,CY + 2: POKE 35,CY + 14
360 M$ = "": HTAB 1: VTAB 10
361 PH =  PEEK (1403) + 1:PV =  PEEK (37) + 1: I
     NVERSE : PRINT " ";: NORMAL : HTAB PH: VTA
     B PV: CALL TCHE,R
362  IF R = 27 THEN  RETURN
363  IF R = 8 OR R = 127 THEN  GOSUB 390: GOTO 3
     61
364  IF R = 13 AND M$ <  > "" THEN  PRINT " ";:
     GOTO 450
365  IF R < 46 OR (R > 57 AND R < 65) OR (R > 90
      AND R < 97) OR R > 122 THEN  POKE 1403, P
     EEK (1403) - 1: GOTO 361
370  PRINT  CHR$ (R);
375 M$ = M$ +  CHR$ (R)
380  GOTO 361
390  HTAB PH: VTAB PV: PRINT " ";: IF  LEN (M$)
     = 0 THEN  PRINT  CHR$ (7);: POKE 1403, PEE
     K (1403) - 2: RETURN
391  IF  LEN (M$) <  = 1 THEN  POKE 1403, PEEK (
     1403) - 2:M$ = "": RETURN
392  POKE 1403, PEEK (1403) - 2:M$ =  LEFT$ (M$,
      LEN (M$) - 1): RETURN
450  PRINT D$"BSAVE"M$",A$2000,L$3FFF"
480  IF ER <  > 0 THEN ER = 0: GOTO 330
490  RETURN
499 :
500  REM  DESSIN D'UN CADRE
510 :
520  POKE 32,CX + 1: POKE 33,CX + 60: POKE 34,CY
     : POKE 35.CY + 15: HOME : TEXT
530  HTAB CX + 5: VTAB CY + 1: PRINT IT$
540  VTAB CY: HTAB CX + 2: FOR I = CX + 2 TO CX
     + 27: PRINT "_";: NEXT
550  VTAB CY + 15: HTAB CX + 1: FOR I = CX + 1 T
     O CX + 60: PRINT "_";: NEXT : PRINT "!"
560  VTAB CY + 1: HTAB CX + 28: PRINT "!";: FOR
     I = CX + 29 TO CX + 60: PRINT "_";: NEXT
570  FOR I = CY + 1 TO CY + 15: POKE 36,CX: VTAB
      I: PRINT "!": NEXT
580  FOR I = CY + 2 TO CY + 15: POKE 36,CX + 60:
      VTAB I: PRINT "!": NEXT
```

```
590  RETURN
599 :
790  REM  SORTIE DU PROGRAMME
800  HTAB 1: VTAB 23: TEXT : CALL  - 868: PRI
     NT "Avez-vous vraiment décidé d'arreter
      (O/N) ? ";: GET R$
801 :
810  IF R$ <  > "O" AND R$ <  > "o" THEN  RET
     URN
820  HOME : END
890 :
900  REM  SAUVEGARDE D'UNE IMAGE SUR DISK
905 :
910  POKE 49236,0
920  POKE 49152,0: CALL TRANSFERT,0,1: POKE 4
      9153,0: REM  TRANSFERT MEM AUXI EN PAGE
      2 PRINCIPALE
930 IT$ = "Sauvegarde":NM = 2:FL = 1: GOSUB 3
     00
955 FL = 0
960  RETURN
990 :
1000  REM  AFFICHAGE MENU
1005 :
1010 C1 = 1:CX = CX(NM):CY = CY(NM):IT$ = IT$
     (IM): GOSUB 500:HT = HT(NM):S1 = S(IM):H
     L = HL(NM)
1015  HTAB 1: VTAB 24: PRINT "Choisir avec le
     s fleches / Valider par <ENTER> / Quitte
     r par <ESC>                ";
1020 C1 = 1
1030  FOR X = 1 TO S1: IF X = C1 THEN  INVERS
     E
1040  VTAB HL + 2 + X + X * (S1 < 10): HTAB H
     T: PRINT S$(IM,X): NORMAL
1050  NEXT X
1070  POKE  - 16368,0
1080  CALL TCHE,K
1090  IF K > 48 AND K < 48 + S1 THEN C1 = K:
     GOTO 1170
1100  IF K <  > 8 AND K <  > 21 AND K <  > 13
      AND K <  > 10 AND K <  > 11 AND K <  >
     27 THEN 1080
1140 O1 = C1
1150  IF K = 8 OR K = 11 THEN C1 = C1 - 1:C1
     = C1 * (C1 > 0) + S1 * (C1 = 0): GOSUB 1
     180: GOTO 1080
1160  IF K = 21 OR K = 10 THEN C1 = C1 + 1:C1
      = C1 * (C1 <  = S1) + (C1 = S1 + 1): GO
     SUB 1180: GOTO 1080
1170  RETURN
1180  VTAB HL + 2 + O1 + O1 * (S1 < 10): HTAB
      HT: PRINT S$(IM,O1)
1190  VTAB HL + 2 + C1 + C1 * (S1 < 10): INVE
     RSE : HTAB HT: PRINT S$(IM,C1): NORMAL :
      RETURN
1200 :
4990 :
5000  REM  HARD COPY
5005 :
5007  HTAB 1: VTAB 24: PRINT " Entrer la nouv
     elle valeur / Valider par <ENTER> / Quit
     ter par <ESC> ";
5010 CX = CX(NM):CY = CY(NM):IT$ = "Hard copy
     ": GOSUB 500
5020  POKE 768,0: POKE 769,0: REM  IMAGE NORM
     ALE ET 560*192
5030  POKE 32,CX + 4: VTAB CY + 4: PRINT
5040 W$ = P1$: PRINT "ECHELLE HORIZONTALE (1/
     2) ?";: GOSUB 5300: IF W <  > 13 THEN P1
     $ = W$: POKE 36,HT + 15: VTAB VT + 1: PR
     INT P1$: PRINT
5050 W$ = P2$: PRINT "ECHELLE VERTICALE   (1/
     2) ?";: GOSUB 5300: IF W <  > 13 THEN P2
     $ = W$: POKE 36,HT + 15: VTAB VT + 1: PR
     INT P2$: PRINT
5070  IF P1$ = "2" AND P2$ = "1" THEN  FOR I
     = 0 TO 4: POKE 4423 + I,234: NEXT :P3$ =
      "002": GOTO 5140
5080  IF P1$ = "1" AND P2$ = "2" THEN  POKE 7
     68,128: FOR I = 5 TO 7: POKE 4423 + I,23
     4: NEXT
5090  IF P1$ = "2" AND P2$ = "2" THEN  POKE 7
     68,128:P3$ = "002": GOTO 5140
5100 W$ = P3$
5110  PRINT "MARGE GAUCHE ?";:HT =  PEEK (36)
     :VT =  PEEK (37): PRINT P3$;: POKE 36,HT
      + 2: VTAB VT + 1: INPUT "";P3$: PRINT P
     3$;
5120  IF P3$ = "" THEN P3$ = W$
5130 P3$ =  RIGHT$ ("000" + P3$,3)
5140  CALL ACTIVE
5150  PRINT  CHR$ (27) +  CHR$ (62);: PRINT
     CHR$ (27) +  CHR$ (76) + P3$;
5160  CALL HARDCOPY
5185  POKE 4423,44: POKE 4424,0: POKE 4425,3:
      POKE 4426,16: POKE 4427,3: POKE 4428,32
     : POKE 4429,237: POKE 4430,253
5190  TEXT : HOME
5200  RETURN
5300 HT =  PEEK (36):VT =  PEEK (37): PRINT W
     $;
5305  POKE 36,HT + 15: VTAB VT + 1: CALL TCHE
     ,W
5310  IF W = 27 THEN  POP : RETURN
5312  IF W = 13 THEN 5320
5315  IF (W <  > 49 AND W <  > 50) THEN  PRIN
     T  CHR$ (7);: GOTO 5305
5320 W$ =  CHR$ (W): PRINT W$: RETURN
8999 :
9000  REM  MENU PRINCIPAL
9010 :
9020  TEXT : HOME
9030  PRINT "       IMAGES DOUBLE HAUTE RESOLU
     TION     -     R.Jost (1986)"
9040 NM = 1:IM = 1
9050  GOSUB 1000
9055  IF K = 27 THEN  GOSUB 800: GOTO 9020
9060 NM = 2: ON C1 GOSUB 10000,12000,900,5000
     ,800
9070  GOTO 9020
9990 :
10000  REM  MENU CHARGEMENT
10010 :
10020 IM = 2: GOSUB 1010
10030  IF K = 27 THEN  RETURN
10040 NM = 3:C0 = C1:TA$ = " 17": IF C0 = 1 T
     HEN TA$ = " 33"
10050  ON C1 GOSUB 10070,10080,10120,10130: T
     EXT : HOME
10060  RETURN
10070  POKE 49236,0:IT$ = "Image DHGR 16K":A$
      = ",A$2000": GOSUB 100: POKE 49152,0: C
     ALL TRANSFERT,1,0: POKE 49153,0: GOSUB 1
     2070: POKE 49236,0: RETURN
10080 IT$ = "deux moitiés DHGR"
```

```
10090  POKE 49236,0: GOSUB 100:A$ = ",A$2000"
     : POKE 49237,0: GOSUB 260
10100  POKE 49236,0: GOSUB 181: GOSUB 260: GO
     SUB 12070
10110  RETURN
10120  POKE 49236,0:IT$ = "deux images HGR":A
     $ = ",A$4000": GOSUB 100: CALL H1,0: GOS
     UB 181: CALL H1,20: GOSUB 12070: RETURN
10130 IT$ = "image HGR -> DHGR 16K":A$ = ",A$
     2000": GOSUB 100: CALL H2: GOSUB 12070:
     RETURN
11990 :
12000  REM  TRAITEMENT IMAGE
12010 :
12020 IM = 3:NM = 2
12030  GOSUB 1010
12040  IF K = 27 THEN  RETURN
12045  IF C1 = 1 THEN  GOSUB 12070: GOSUB 107
     0: GOTO 12040
12050  ON C1 GOSUB 12070,12080,12100,12110,12
     120,12130,12090,12150,12170,12140,12160,
     12180: GOSUB 12070: GOSUB 1070
12060  GOTO 12040
12070  POKE 49160,0: POKE 49165,0:X =  PEEK (
     49232) +  PEEK (49234) +  PEEK (49239) +
      PEEK (49246): REM  AFFICHE L'IMAGE
12072 R =  PEEK ( - 16384): ON R < 128 GOTO 1
     2072:R = R - 128: POKE  - 16368,0
12074  IF R = 13 OR R = 27 THEN  TEXT : RETUR
     N
12075  GOTO 12072
12080  HGR : POKE 49234,0: POKE 49237,0: CALL
      62450: POKE 49236,0: RETURN : REM  EFFA
     CE L'IMAGE
12090  CALL NEG: TEXT : RETURN : REM  NEGATIF
      DE L'IMAGE
12100  CALL TRANSFERT,3,2: TEXT : RETURN : RE
     M   TRANSFERT EN MEM AUXI
12110  CALL TRANSFERT,5,2: TEXT : RETURN : RE
     M   VERS MEM AUXI $8000-BFFF
12120  CALL TRANSFERT,2,2: TEXT : RETURN : RE
     M   RAPPEL DE LA MEM AUXI
12130  CALL TRANSFERT,4,2: TEXT : RETURN : RE
     M   RAPPEL AUXI $8000-BFFF
12140  CALL CG: RETURN : REM  COPIE GAUCHE ->
      DROITE
12150  CALL IG: RETURN : REM  INVERSE PARTIE
     GAUCHE
12160  CALL MI: RETURN : REM   MIROIR
12170  CALL ECHANGE: RETURN : REM  PERMUTER L
     ES DEUX MOITIES DE L'ECRAN DHGR
12180  POP : RETURN
14990 :
15000  REM  TRAITEMENT ERREUR DOS
15005 :
15010  CALL FAUTE
15020  PRINT  CHR$ (7);: HTAB 1: VTAB 15
15030 ER =  PEEK (222):EL =  PEEK (218) + 256
      *  PEEK (219)
15040  INVERSE
15050  IF ER = 4 THEN  PRINT "DISK PROTEGE EN
      ECRITURE";: GOTO 15120
15060  IF ER = 6 AND EL = 20090 THEN  PRINT "
     METTRE UNE DISQUETTE CONTENANT DHGRTOOL.
     0": CALL  - 741: GOTO 20090
15070  IF ER = 6 AND EL = 20095 THEN  PRINT "
     LE PROGRAMME HARD (POM'S #18) DOIT FIGUR
     ER SUR LA DISQUETTE": CALL  - 741: GOTO
     20100
15080  IF ER = 6 THEN  PRINT "CE FICHIER N'EX
     ISTE PAS";: GOTO 15120
15090  IF ER = 8 THEN  PRINT "ERREUR D'ENTREE
     /SORTIE ! ";: GOTO 15120
15100  IF ER = 9 THEN  PRINT "DISK REMPLI ";:
      PRINT D$"DELETE"M$: GOTO 15120
15110  PRINT "ERREUR #"ER" A LA LIGNE #"EL: T
     EXT : END
15120  NORMAL : FOR T = 1 TO 2000: NEXT : PRI
     NT  CHR$ (7);
15130  IF EL = 450 OR EL = 460 THEN 480
15140  END
19990 :
20000  REM
20005 :
20010  TEXT : HOME : ONERR  GOTO 15020
20040 D$ =  CHR$ (4): PRINT D$"PR#3"
20070 TRANSFERT = 2048:NEG = TR + 3:MI = TR +
      6::H2 = TR + 9:H1 = TR + 12:EC = TR + 1
     5:CG = TR + 18:IG = TR + 24:FAUTE = TR +
      30:HARDCOPY = 4144
20080 TCHE = TR + 33:ACTIVE = TR + 36:DESACTI
     VE = TR + 39
20090  IF  PEEK (2048) +  PEEK (2050) <  > 86
      THEN  PRINT D$"BLOAD DHGRTOOL.0"
20095  IF  PEEK (4144) +  PEEK (4154) <  > 38
     5 THEN  PRINT D$"BLOAD HARD,A$1000"
20096  POKE 4304,16: POKE 4334,16: POKE 4339,
     16: POKE 4443,16: POKE 4453,16: POKE 448
     7,16
20097  POKE 4151,234: POKE 4152,234: POKE 415
     3,234: POKE 4506,DESACTIV / 256: POKE 45
     05,DESACTIV -  PEEK (4506) * 256: REM
     MODIFIE LES APPELS PR# DANS HARD.
20100  DIM S$(5,14)
20105  ONERR  GOTO 15000
20110 P1$ = "1":P2$ = "1":P3$ = "002"
20120  DATA     "Charger une image",    "Trait
     er l'image",    "Sauver une image DHGR",
      "Imprimer une image DHGR",    "Quitter"

20130 S(1) = 5: FOR I = 1 TO S(1): READ S$(1,
     I): NEXT :HT(1) = 25:HL(1) = 2:CX(1) = 5
     :CY(1) = 2:IT$(1) = "Menu Principal"
20140  DATA     "Charger une image 16K",    "C
     harger deux images DHGR",    "Charger deu
     x images HGR",    "Charger une image HGR"
     ,    "Retour au MENU principal"
20150 S(2) = 5: FOR I = 1 TO S(2): READ S$(2,
     I): NEXT :HT(2) = 28:HL(2) = 4:CX(2) = 7
     :CY(2) = 4:IT$(2) = "Chargement d'une im
     age"
20160  DATA      "Voir l'image",     "Effacer
     l'image",     "Sauver en mem auxi-1",
     "Sauver en mem auxi-2",     "Rappel de la
      mem auxi-1",     "Rappel de la mem auxi-
     2"
20170  DATA      "Negatif de l'image",     "Ne
     gatif moitie gauche",     "Permute les de
     ux moities",     "Copie moitie gauche a d
     roite",     "Miroir",     "Retour au MENU
     principal"
20180 S(3) = 12: FOR I = 1 TO S(3): READ S$(3
     ,I): NEXT :HT(3) = 31:HL(3) = 3:CX(3) =
     9:CY(3) = 6:IT$(3) = "Traitement image"
20210 X =  PEEK (49239): NORMAL : GOTO 9000
```

Récapitulation DHGRTOOL.0

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :
BSAVE DHGRTOOL.C,A$800,L$39B

```
0800- 4C B7 0A 4C 2A 08 4C 50
0808- 08 4C A2 08 4C EC 08 4C
0810- 8F 09 4C DB 09 4C 1F 0A
0818- 4C 65 0A 4C 78 0A 4C 47
0820- 0B 4C 75 0B 4C 51 0B 4C
0828- 68 0B EA A2 00 86 06 A2
0830- 20 86 07 A0 00 8D 55 C0
0838- B1 06 49 FF 91 06 8D 54
0840- C0 B1 06 49 FF 91 06 88
0848- D0 EB E6 07 CA D0 E4 60
0850- EA A2 00 86 EB 86 ED 86
0858- 06 A2 BF 86 07 86 EE A2
0860- 28 86 EC A6 06 20 4A 09
0868- 85 18 84 19 A6 07 20 4A
0870- 09 85 1A 84 1B A4 EB 8D
0878- 55 C0 B1 18 48 B1 1A 91
0880- 18 68 91 1A 8D 54 C0 B1
0888- 18 48 B1 1A 91 18 68 91
0890- 1A C8 C4 EC D0 E1 E6 06
0898- C6 07 38 A5 07 E5 06 B0
08A0- C2 60 EA A2 00 86 06 A2
08A8- 20 86 07 A0 00 B1 06 86
08B0- 09 20 CD 08 A6 09 8D 55
08B8- C0 A5 50 91 06 8D 54 C0
08C0- A5 51 91 06 88 D0 E6 E6
08C8- 07 CA D0 DF 60 A2 00 86
08D0- 50 86 51 A2 07 0A 08 20
08D8- E7 08 28 20 E7 08 CA 10
08E0- F4 20 E7 08 46 50 60 26
08E8- 50 26 51 60 EA A0 00 84
08F0- EB 84 ED A0 28 84 EC A0
08F8- BF 84 EE 20 F5 E6 86 1C
0900- 86 1D A6 EE 20 4A 09 85
0908- 18 98 18 69 20 85 19 20
0910- 4A 09 85 1A 84 1B A4 EB
0918- B1 18 48 C8 B1 18 C8 84
0920- EB A4 1D 8D 54 C0 91 1A
0928- 68 8D 55 C0 91 1A E6 1D
0930- 8D 54 C0 A5 EB C5 EC D0
0938- DD A9 00 85 EB A5 1C 85
0940- 1D C6 EE A5 EE C9 FF D0
0948- B9 60 8A 4A 4A 4A A8 B9
0950- 5F 09 48 8A 29 07 0A 0A
0958- 18 79 77 09 A8 68 60 00
0960- 80 00 80 00 80 00 80 28
0968- A8 28 A8 28 A8 28 A8 50
0970- D0 50 D0 50 D0 50 D0 20
0978- 20 21 21 22 22 23 23 20
0980- 20 21 21 22 22 23 23 20
0988- 20 21 21 22 22 23 23 A2
0990- 00 86 EB 86 ED A2 C0 86
0998- EE A2 14 86 EC A6 ED 20
09A0- 4A 09 85 18 84 19 A4 EB
09A8- 8D 55 C0 20 C2 09 8D 54
09B0- C0 20 C2 09 C8 C4 EC D0
09B8- EF E6 ED A5 ED C5 EE D0
09C0- DC 60 B1 18 85 1C 98 48
09C8- 18 69 14 A8 B1 18 85 1D
09D0- A5 1C 91 18 68 A8 A5 1D
09D8- 91 18 60 A2 00 86 EB 86
09E0- ED A2 C0 86 EE A2 14 86
09E8- EC A6 ED 20 4A 09 85 18
09F0- 84 19 A4 EB 8D 55 C0 20
09F8- 0E 0A 8D 54 C0 20 0E 0A
0A00- C8 C4 EC D0 EF E6 ED A5
0A08- ED C5 EE D0 DC 60 B1 18
0A10- 85 1C 98 48 18 69 14 A8
0A18- A5 1C 91 18 68 A8 60 A2
0A20- 14 86 EB A2 00 86 ED A2
0A28- C0 86 EE A2 28 86 EC A6
0A30- ED 20 4A 09 85 18 84 19
0A38- A4 EB 8D 55 C0 20 54 0A
0A40- 8D 54 C0 20 54 0A C8 C4
0A48- EC D0 EF E6 ED A5 ED C5
0A50- EE D0 DC 60 B1 18 85 1C
0A58- 98 48 38 E9 14 A8 A5 1C
0A60- 91 18 68 A8 60 A2 00 86
0A68- EB A2 00 86 ED A2 C0 86
0A70- EE A2 14 86 EC 4C 8B 0A
0A78- A2 14 86 EB A2 00 86 ED
0A80- A2 C0 86 EE A2 28 86 EC
0A88- 4C 8B 0A A6 ED 20 4A 09
0A90- 85 18 84 19 A4 EB 8D 55
0A98- C0 20 B0 0A 8D 54 C0 20
0AA0- B0 0A C8 C4 EC D0 EF E6
0AA8- ED A5 ED C5 EE D0 DC 60
0AB0- B1 18 49 FF 91 18 60 EA
0AB8- 20 F5 E6 86 06 20 F5 E6
0AC0- E0 00 D0 04 20 DB 0A 60
0AC8- E0 01 D0 04 20 F8 0A 60
0AD0- E0 02 D0 06 20 DB 0A 20
0AD8- F8 0A 60 A6 06 8D 54 C0
0AE0- 20 18 0B BD 23 0B 85 3D
0AE8- BD 29 0B 85 3F BD 2F 0B
0AF0- 85 43 8A 6A 20 11 C3 60
0AF8- A6 06 20 18 0B BD 35 0B
0B00- 85 3D BD 3B 0B 85 3F BD
0B08- 41 0B 85 43 8D 55 C0 8A
0B10- 6A 20 11 C3 8D 54 C0 60
0B18- A9 00 85 3C 85 42 A9 FF
0B20- 85 3E 60 20 40 40 20 80
0B28- 20 3F 5F 5F 3F 9F 3F 40
0B30- 20 20 40 20 80 20 20 60
0B38- 20 A0 20 3F 3F 7F 3F BF
0B40- 3F 40 40 20 60 20 A0 68
0B48- A8 68 A6 DF 9A 48 98 48
0B50- 60 AD 30 BE 8D FE 02 AD
0B58- 31 BE 8D FF 02 A9 00 8D
0B60- 30 BE A9 C1 8D 31 BE 60
0B68- AD FE 02 8D 30 BE AD FF
0B70- 02 8D 31 BE 60 20 BE DE
0B78- 20 E3 DF AD 00 C0 10 FB
0B80- 2C 10 C0 C9 FF F0 06 C9
0B88- E0 90 02 29 DF 29 7F A8
0B90- 20 01 E3 A6 83 A4
```

Récapitulation HARD

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :
BSAVE HARD, A$4000,L$1A1

```
4000- 00 20 80 20 00 21 80 21
4008- 00 22 80 22 00 23 80 23
4010- 28 20 A8 20 28 21 A8 21
4018- 28 22 A8 22 28 23 A8 23
4020- 50 20 D0 20 50 21 D0 21
4028- 50 22 D0 22 50 23 D0 23
4030- D8 78 8D 54 C0 A9 01 20
4038- 95 FE A9 89 20 ED FD A9
4040- DA 20 ED FD A9 8D 20 ED
4048- FD A9 9B 20 ED FD A9 D4
4050- 20 ED FD A9 B1 20 ED FD
4058- A9 B6 20 ED FD AD 5E C0
4060- AD 50 C0 AD 57 C0 AD 52
4068- C0 8D 0D C0 8D 01 C0 8D
4070- 04 C0 8D 02 C0 A9 00 8D
4078- 0A 03 8D 11 03 A9 FF 8D
4080- 10 03 2C 00 03 10 10 AD
4088- 10 03 49 FF 8D 10 03 2C
4090- 10 03 10 03 CE 0A 03 A9
4098- 8D 20 ED FD A9 9B 20 ED
40A0- FD A9 F0 20 ED FD A9 FF
40A8- 8D 0B 03 A9 00 8D 0C 03
40B0- A9 9B 20 ED FD A9 C7 20
40B8- ED FD A9 B1 20 ED FD A9
40C0- B1 20 ED FD A9 B2 20 ED
40C8- FD A9 B0 20 ED FD 4C D1
40D0- 40 A9 FF EE 0B 03 4D 0C
40D8- 03 8D 0C 03 8D 55 C0 D0
40E0- 06 8D 54 C0 CE 0B 03 AD
40E8- 0A 03 0A AA BD 00 40 85
40F0- 18 BD 01 40 2C 00 03 10
40F8- 08 2C 10 03 10 03 18 69
4100- 10 85 19 A2 00 AC 0B 03
4108- A9 00 8D 11 03 B1 18 9D
4110- 02 03 E8 E0 08 F0 18 2C
4118- 00 03 10 0A AD 11 03 49
4120- FF 8D 11 03 D0 E7 A9 04
4128- 18 65 19 85 19 D0 DE A0
4130- 07 A2 08 5E 01 03 2A CA
4138- D0 F9 8D 54 C0 2C 01 03
4140- 10 02 49 FF 20 ED FD 2C
4148- 00 03 10 03 20 ED FD 88
4150- D0 DF A9 27 CD 0B 03 F0
4158- 03 4C D1 40 A9 00 CD 0C
4160- 03 F0 03 4C D1 40 EE 0A
4168- 03 A9 03 85 FF A9 30 85
4170- FE A9 00 20 ED FD C6 FE
4178- D0 F7 C6 FF D0 F3 A9 18
4180- CD 0A 03 F0 03 4C 82 40
4188- 8D 54 C0 8D 51 C0 8D 00
4190- C0 A9 8D 20 ED FD A9 03
4198- 20 95 FE A9 8D 20 ED FD
41A0- 60
```

Un jeu de réflexion : Souris-Laser

//e+ //c //gs	ProDOS

Benoît Mesnil

Souris-Laser vous est présenté ici en deux versions : Basic Applesoft et assembleur. Une occasion de comparer la conception et d'apprécier la vitesse de réaction... peut-être aussi d'adopter définitivement l'assembleur.

Souris Laser Version Basic

La souris livrée avec les //c et adaptable aux //e est un accessoire fort agréable pour l'utilisation de nombreux logiciels. La documentation de base ne permet pas d'en maîtriser la programmation de haut niveau mais indique un moyen simple de la contrôler sous BASIC. La routine correspondante est ici mise en œuvre pour la manipulation d'un jeu de réflexion généralement connu sous le nom de "boîte noire".

Les règles

Aux intersections d'une grille 8 x 8, trois atomes se placent au hasard. En émettant, horizontalement ou verticalement, des rayons 'laser' à partir des canons disposés à la périphérie de la grille, et en observant les points de sortie de ces rayons, il faut trouver l'emplacement des atomes sachant que :

- un rayon frappant directement un atome est absorbé ;
- tout rayon détectant un atome en avant sur sa droite (respectivement gauche) est dévié sur la gauche (respectivement droite) ;
- un rayon dont la trajectoire passe entre deux atomes situés immédiatement de part et d'autre est réfléchi ;
- si un atome est situé contre un bord, tout rayon émis d'un canon disposé de part et d'autre de l'atome est immédiatement réfléchi ;
- les canons ont le double rôle d'émetteur et de détecteur ;
- le tir est déclenché en cliquant la souris dans l'un ou l'autre des canons (zones jaunes extérieures à la grille) ; trois tirs sont possibles sur chacun.

Au lancement du programme (`RUN SOURIS.LASER`), un menu d'options constitué de grosses lettres en basse résolution s'affiche. Cliquer dans la case correspondante pour choisir :

Jeu normal : le premier curseur est rose ; on dispose de 6 coups.

Jeu facile : le premier curseur est vert ; on dispose de 12 coups et les trajectoires des rayons sont visualisées.

Démonstration : les atomes sont visibles et les trajectoires visualisées ; on peut vérifier les règles du jeu.

Stop : fin du jeu.

Une fois l'option principale sélectionnée, le jeu peut commencer. Un second menu d'options est affiché, les deux premières dans la couleur du niveau choisi (vert ou rose) :

- commencer une nouvelle partie au même niveau de difficulté ;
- continuer la partie en cours, du moins s'il reste des couleurs ;
- donner les solutions en affichant les atomes.

Stop : fin du jeu.

La signification des initiales est rappelée dans un message en bas de l'écran.

Pour effectuer le tir, on amène le curseur sur un canon et on 'clique' ; un pavé de la couleur du curseur s'inscrit dans le canon d'origine puis, sauf absorption, dans le détecteur de sortie.

On peut à tout moment tester la présence d'un atome : amener le curseur dans la grille et cliquer à l'emplacement supposé. Si un atome s'y trouve effectivement, un pavé vert pâle s'inscrit et un signal retentit. Pour reprendre le jeu, ramener le curseur hors de la grille ou dans la case de l'option C et cliquer. Le pointeur reprend la couleur du tir suivant ou reste noir s'il n'y a plus de couleur disponible. Il n'y a pas de pénalité en cas de test infructueux d'une solution : ceci peut être envisagé à condition d'éliminer les cas d'échec dus à un positionnement imprécis du pointeur.

On pourrait augmenter à quatre le nombre d'atomes ; paradoxale-

ment, cela semble rendre le jeu plus facile.

Le programme

Voici quelques indications sur la programmation du jeu.

Lignes 30-36 : la souris est mise en service.
40-85 : la grille et le premier menu sont affichés.
100-130 : lecture de la souris: position et déclic.
135-138 : temporisation du curseur piloté par la souris.
140-180 : la souris a été 'cliquée', aiguille vers le traitement approprié.
200 : souris cliquée dans la zone des menus.
205-248 : si le premier menu affiché, enregistre l'option et appelle le second menu.
250-280 : traite l'option du deuxième menu.
400-520 : le rayon a été émis ou se propage horizontalement:
460-465 : en ligne supérieure de la grille.
470-475 : en ligne inférieure de la grille.
480-488 : jeu facile, affiche un traceur.
494-498 : le rayon ressort à droite ou à gauche.
500-520 : il tourne ou ressort en haut ou en bas.
600-705 : le rayon a été émis ou se propage verticalement. On retrouve, au chiffre des centaines près, les groupes de ligne effectuant les traitements équivalents à ceux des sections listées ci-dessus.
720-726 : fin de tir.
900-910 : fin du jeu, déconnecte la souris.

Sous-programmes

750-790 : place les atomes.
800-845 : donne les solutions.
850-890 : la souris a été cliquée dans la grille : teste les solutions.

Ayant été un de mes tout premiers programmes sur Apple, celui-ci n'est pas optimisé et est de ce fait assez lent ; les indications ci-dessus permettront de mener à bien les améliorations nécessaires.

Laser.Mouse version 'Asm'

Le jeu obéit aux mêmes règles que la version Basic et se présente de façon tout-à-fait similaire : les options sont choisies en cliquant dans des cases de menu dessinées en basse-résolution :

Menu 1

J Jeu normal, 6 essais, pas de visualisation des trajectoires ;

F Jeu facile, 12 essais, visualisation des trajectoires.

D Démonstration, comme F, avec visualisation des cibles.

Menu 2

J Rejouer ; réinitialisation et retour au Menu 1.

C Continuer ; il n'est toutefois plus nécessaire d'y cliquer pour reprendre le jeu après avoir recherché une cible dans la grille.

D Donne les solutions.

Dans les deux cas, **S** termine le jeu.
La principale différence vient évidemment de la rapidité d'exécution qui ne laisse aucune chance de situer les obstacles par la durée de l'essai.
Malgré les temporisations, il est même recommandé de ne pas s'endormir sur le bouton de la souris...
Un problème demeure qui, apparemment, n'est pas d'origine logicielle : il faut veiller à ne pas déplacer la souris au moment où l'on clique, en particulier dans la case **J** du Menu 1.

Le jeu se compose de deux éléments principaux :
- Le fichier `LASER.MOUSE`, chargé en $6000 : programme de jeu proprement dit.
- Le fichier `LASER.SBR`, chargé en $7000 : sous-routines et utilitaires. Il convient de les charger (BLOAD) dans cet ordre avant de démarrer par un CALL 24576. Le programme Basic `LASER.START` se charge de ces opérations.

Descriptif

Comme pour la version Basic, voici quelques informations sur la conception du programme.

Mémoires Page Zéro

$6-7 : adresse de base des *firmwares* souris ($C400 pour slot 4): on a en effet utilisé le mode standard d'appel des sous-programmes souris via les vecteurs de la table fournie par le constructeur.
$8-9 : adresse des tables de messages.
$19-1A : adresse ($7600 actuellement) de la grille réduite 8 x 8 dans laquelle s'exécute le jeu.
$1B-1C : coordonnées x et y des particules en mouvement dans cette grille.
$1 : drapeau initialisé à 0 ; vaut 1 pour jeu normal, 3 pour jeu facile et 5 pour le mode démonstration. Bit 7 à 0 pour déplacement vers la droite ou vers le bas, à 1 pour déplacement vers la gauche ou vers le haut.
$1E : couleur du curseur en cours.
$1F : sauvegarde de la couleur du fond sous le curseur.
$4E-4F : RNDL et RNDH utilisés pour positionnement 'au hasard' des cibles.
$FE-FF : coordonnées x et y du curseur à l'écran.

Routines système

Le programmes utilise les routines souris via les vecteurs de la table en page $C4.
Il utilise également les routines suivantes :

PLOT	$F800
HLINE	$F819
VLINE	$F828
CLEARTOP	$F836
SETCOL	$F864
SCRN	$F871
SETTEXT	$FB39
SETGR	$FB40
HOME	$FC58
BELL	$FBDD
WAIT	$FCA8
COUT	$FDED

Programme LASER.MOUSE

$6000-600E : Initialisation d'adresses
$6010-6017 : INITMOUSE (appelé via CallMouse en $7000)

$6018-6030 : CLAMP X (id.)
$6031-6036 : CLAMP Y (id.)
$6037-6040 : SETMOUSE (id.). Mode transparent adopté par facilité.

$6041-6048 : CLEARMOUSE (id.)
$6049 : Appel de Menu1 en $700B
$604C : Mise à zéro de la grille de jeu et tirage des cibles par les sous- routines en $700B et $7138
$6050-607F : READMOUSE: lecture de la souris et affichage/ déplacement du curseur ; boucle tant que la souris n'est pas cliquée.

Au-delà de $6080, traitement selon les choix obtenus par "clics":
$6089-6094 et au-delà de $60EB: on a cliqué dans la zone de jeu.
$6096-60CA : stop ou options premier menu.
$60CB-60EA : options deuxième menu.
$60EB-6104 : élimination des coups illégaux.
$6105-612D : *dispatche* vers tir en ligne ($615A) ou en colonne ($6220).
$612E-6157 : on a cliqué dans la grille ; recherche de solution.

Tir en ligne

$615A-6183 : début de tir ; tests sur les bords de la grille.
$6184-61F9 : la particule se déplace horizontalement.
$61FB-6205 : la particule a été réfléchie.
$6206-6215 : la particule sort horizontalement.

Tir en colonne

$6220-625F : début de tir.
$6260-62E3 : déplacement vertical de la particule.
$62E4-62EE : la particule a été réfléchie.
$62EF-62FB : la particule sort verticalement.
$6300-6323 : fin de tir ; test sur la couleur ; temporisation ($631C-$6320) et retour à READMOUSE en $6050.
$6330-6340 : fin du jeu ; déconnexion de la souris.

Sous-programmes LASER.SBR

$7000-700A : *dispatche* vers les routines firmware souris.
$700B-7137 : affichage du premier menu : traçage des cases ($7011-704F), des lettres ($7050-70DB), de la grille ($70DC-7119) ; affichage des premiers messages ($711A-7137); la routine d'affichage en 7126-7137 est également utilisée pour les autres messages.
$7138-717E : Mise à zéro de la grille de jeu ($7138-7142) et tirage des positions de cibles par transformations de RNDH et RNDL.
$7180-7185 : transforme les coordonnées écran en coordonnées dans la grille de jeu.
$7188-718C : transformation inverse.
$7190-719A : positionnement dans la grille de jeu en fonction du rang de la colonne.
$71A0-71DC : affichage du deuxième menu et des messages correspondants.
$71E0-71E9 : affichage du message d'erreur.
$71F0-7224 : visualisation des cibles (options D des cases menus).
$7228-725E : début et fin de tir en ligne ; la section $7246-725E est aussi utilisée en cas de réflexion.
$7260-7290 : début et fin de tir en colonne ; point d'entrée réflexion en $7278.
$7292-72BC : affichage de la trajectoire des particules (options F et D du menu 1).
$72C0-72C6 : changement de signe du drapeau $1D lors des inversions de sens de déplacement.

Programme SOURIS.LASER

```
5  REM  ** LA SOURIS LASER **
6  REM
15 D$ =  CHR$ (4):SC = 25.575
20  DIM PT(7,7)
30  REM  *ALLUME LA SOURIS *
32  PRINT D$;"PR# 4": PRINT  CHR$ (1)
34  PRINT D$;"PR# 0"
36  PRINT D$;"IN# 4"
40  GR : HOME : COLOR= 13
42  VLIN 2,33 AT 2: VLIN 2,33 AT 8
44  HLIN 2,8 AT 2: HLIN 3,7 AT 10: HLIN 3,7
     AT 18
45  HLIN 3,7 AT 26: HLIN 2,8 AT 34
48  COLOR= 11: VLIN 4,8 AT 6: HLIN 4,5 AT 8:
     PLOT 4,7
50  COLOR= 15: FOR X = 28 TO 32 STEP 2: HLIN
     4,6 AT X: NEXT X
51  PLOT 4,29: PLOT 6,31
53  COLOR= 1: HLIN 3,6 AT 20: VLIN 21,23 AT
     4: HLIN 3,6 AT 24: VLIN 21,23 AT 6
70  REM  * TRACE LA GRILLE *
72 CS = 0:CG = 13:CJ = 12:FAC = 0:DEM = 0:PC
     = 2:TD = 15
74  COLOR= CG:ECR = 1
76  FOR X = 14 TO 28 STEP 2
77  VLIN 10,12 AT X: VLIN 14,28 AT X: VLIN 3
     0,32 AT X
78  HLIN 10,12 AT X: HLIN 14,28 AT X: HLIN 3
     0,32 AT X
80  NEXT X
82  COLOR= 0: FOR X = 4 TO 6: VLIN 12,16 AT
     X: NEXT X
85  COLOR= 12: VLIN 12,16 AT 4: HLIN 5,6 AT
     12: PLOT 5,14
90 S =  FRE (0)
94 T3$ = " J -> JEU NORMAL   F -> FACILE"
96 T4$ = " D -> DEMONSTRATION  S -> STOP"
100  REM  * Lit la souris *
102  HOME
105  COLOR= CS: PLOT X,Y
110  VTAB 21: PRINT T3$: VTAB 22: PRINT T4$
120  INPUT "";X,Y,S
125 X =  INT (X / SC):Y =  INT (Y / SC):CS =
     SCRN( X,Y)
130  IF S = 1 THEN 150
```

```
135  COLOR= CJ: FOR A = 0 TO 20: PLOT X,Y: N
     EXT A
138  GOTO 105
140  REM  ** SOURIS CLIQUEE _ CHOIX OU TIR *
     *
150  IF X < 8 THEN 200
155  IF X =  > 14 AND X <  = 28 AND Y =  > 1
     4 AND Y <  = 28 AND ECR = 2 THEN 850
160  IF X < 10 OR X > 32 OR Y < 10 OR Y > 32
      OR CS = 0 THEN 105
165  COLOR= CJ
170  IF X < 14 OR X > 28 THEN 400: REM    LI
     GNE
175  IF Y < 14 OR Y > 28 THEN 600: REM    CO
     LONNE
180  GOTO 105
200  IF Y > 26 THEN 900
202  IF ECR = 2 THEN 250
205  IF Y < 10 THEN CJ = 11: GOTO 220
210  IF Y > 10 AND Y < 18 THEN FAC = 1:PC =
     1: GOTO 220
215  IF Y > 18 AND Y < 26 THEN FAC = 1:DEM =
      1:PC = 1:TD = 30
220  GOSUB 750
225  IF DEM = 1 THEN  GOSUB 800
235 T3$ = "C -> CONTINUER, J -> REJOUER"
238 T4$ = "D -> DONNE LES SOLUTIONS"
240  COLOR= 0: FOR X = 4 TO 6: VLIN 12,16 AT
      X: NEXT X
242  COLOR= CJ: VLIN 12,16 AT 4: HLIN 5,6 AT
      12: HLIN 5,6 AT 16
248  HOME :ECR = 2: GOTO 105
250  IF Y < 10 THEN 70
255  IF Y > 10 AND Y < 18 THEN 270
260  IF Y > 18 THEN  GOSUB 800
265  GOTO 105
270  IF CJ > 0 THEN 105
275 T3$ = "IL N'Y A PLUS DE COULEUR"
280  GOTO 105
400 ST = 1:XX = 12:C = 0:I = 0:J = 0
405  IF X > 28 THEN ST =  - 1:XX = 30:C = 7
410 L =  INT ((Y - 14) / 2):YY = 14 + 2 * L
412 A =  SCRN( XX,YY): IF A <  > CG AND A >
     0 THEN XX = XX - ST: GOTO 412
415  PLOT XX,YY: PRINT  CHR$ (7)
420 I = I + 1:J = J + 1
425  IF PT(C,L) = 1 THEN 725
430  IF (C = 0 AND ST < 0) OR (C = 7 AND ST
     > 0) THEN 490
431  IF PT(C + ST,L) = 1 THEN 725
432  IF L = 0 THEN 460
434  IF L = 7 THEN 470
440  IF I = 1 AND (PT(C,L + 1) = 1 OR PT(C,L
      - 1) = 1) THEN C = C - ST:ST =  - ST:
      GOTO 494
445  IF PT(C + ST,L + 1) = 1 AND PT(C + ST,L
      - 1) = 1 THEN ST =  - ST: GOTO 480
450  IF PT(C + ST,L + 1) = 1 THEN I = I + 1:
     ST =  - 1: GOTO 620
452  IF PT(C + ST,L - 1) = 1 THEN I = I + 1:
     ST = 1: GOTO 620
455  GOTO 480
460  REM  Ligne 0
461  IF I = 1 AND PT(C,L + 1) = 1 THEN C = C
      - ST:ST =  - ST: GOTO 494
464  IF PT(C + ST,L + 1) = 1 THEN Y = 12:ST
     =  - 1: GOTO 500
465  GOTO 480
470  REM  Ligne 7
471  IF I = 1 AND PT(C,L - 1) = 1 THEN C = C
      - ST:ST =  - ST: GOTO 494
475  IF PT(C + ST,L - 1) = 1 THEN Y = 30:ST
     = 1: GOTO 500
480  IF FAC = 0 THEN 490
482 XX = 14 + 2 * C:YY = 14 + 2 * L:CO =  SC
     RN( XX,YY)
485  FOR A = 1 TO TD: PLOT XX,YY: NEXT A
488  COLOR= CO: PLOT XX,YY: COLOR= CJ
490 C = C + ST: IF C >  = 0 AND C <  = 7 THE
     N 420
494 X = 14 + 2 * C:Y = 14 + 2 * L
496 A =  SCRN( X,Y): IF A <  > CG AND A > 0
     THEN X = X + ST: GOTO 496
498  GOTO 720
500 X = 14 + 2 * C
505 A =  SCRN( X,Y): IF A <  > CG AND A > 0
     THEN Y = Y + ST: GOTO 505
520  GOTO 720
600 ST = 1:YY = 12:L = 0:J = 0:I = 0: REM  C
     OLONNES
605  IF Y > 28 THEN ST =  - 1:YY = 30:L = 7
610 C =  INT ((X - 14) / 2):XX = 14 + 2 * C
612 A =  SCRN( XX,YY): IF A <  > CG AND A >
     0 THEN YY = YY - ST: GOTO 612
615  PLOT XX,YY: PRINT  CHR$ (7)
620 J = J + 1:I = I + 1
625  IF PT(C,L) = 1 THEN 725
630  IF (L = 0 AND ST < 0) OR (L = 7 AND ST
     > 0) THEN 690
631  IF PT(C,L + ST) = 1 THEN 725
632  IF C = 0 THEN 660
634  IF C = 7 THEN 670
640  IF J = 1 AND (PT(C + 1,L) = 1 OR PT(C -
      1,L) = 1) THEN L = L - ST:ST =  - ST:
      GOTO 694
645  IF PT(C + 1,L + ST) = 1 AND PT(C - 1,L
     + ST) = 1 THEN ST =  - ST: GOTO 680
650  IF PT(C + 1,L + ST) = 1 THEN J = J + 1:
     ST =  - 1: GOTO 420
652  IF PT(C - 1,L + ST) = 1 THEN J = J + 1:
     ST = 1: GOTO 420
655  GOTO 680
660  REM  Colonne 0
661  IF J = 1 AND PT(C + 1,L) = 1 THEN L = L
      - ST:ST =  - ST: GOTO 694
664  IF PT(C + 1,L + ST) = 1 THEN X = 12:ST
     =  - 1: GOTO 700
665  GOTO 680
670  REM  Colonne 7
671  IF J = 1 AND PT(C - 1,L) = 1 THEN L = L
      - ST:ST =  - ST: GOTO 694
674  IF PT(C - 1,L + ST) = 1 THEN X = 30:ST
     = 1: GOTO 700
680  IF FAC = 0 THEN 690
682 XX = 14 + 2 * C:YY = 14 + 2 * L:CO =  SC
     RN( XX,YY)
685  FOR A = 1 TO TD: PLOT XX,YY: NEXT A
688  COLOR= CO: PLOT XX,YY: COLOR= CJ
690 L = L + ST: IF L >  = 0 AND L <  = 7 THE
     N 620
```

```
694 X = 14 + 2 * C:Y = 14 + 2 * L
696 A =  SCRN( X,Y): IF A <  > CG AND A > 0
     THEN Y = Y + ST: GOTO 696
698  GOTO 720
700 Y = 14 + 2 * L
705 A =  SCRN( X,Y): IF A <  > CG AND A > 0
     THEN X = X + ST: GOTO 705
720  PLOT X,Y: PRINT  CHR$ (7): REM  Fin de
     Tir
725 CJ = CJ - PC: IF CJ <  = 0 THEN CJ = 0:T
     3$ = "IL N'Y A PLUS DE COULEUR"
726  GOTO 110
750  REM  * PLACE LES ATOMES *
755  FOR I = 0 TO 7: FOR J = 0 TO 7
760 PT(I,J) = 0: NEXT J,I
770  FOR I = 1 TO 3
775 C =  INT (8 *  RND (1)):L =  INT (8 *  R
     ND (1))
778  IF PT(C,L) = 1 THEN 775
780 PT(C,L) = 1: NEXT I
790  RETURN
800  REM  * Ss-Pr REPONSE *
805  COLOR= 1
810  FOR I = 0 TO 7: FOR J = 0 TO 7
812  IF PT(I,J) = 0 THEN 840
820 Z = 14 + 2 * I:ZZ = 14 + 2 * J
830  PLOT Z,ZZ
840  NEXT J,I
845  RETURN
850  REM  Test des solutions
852  HOME :CS =  SCRN( X,Y)
860  COLOR= CS: PLOT X,Y
864  VTAB 22: PRINT "CLIQUER HORS CADRE POUR
      REJOUER"
866  INPUT "";X,Y,S
870 X =  INT (X / SC):Y =  INT (Y / SC):CS =
      SCRN( X,Y)
872  COLOR= 14: FOR A = 0 TO 20: PLOT X,Y: N
     EXT A
874  IF S <  > 1 THEN 860
876  IF X < 8 AND Y > 26 THEN 900
878  IF X < 14 OR X > 28 OR Y < 14 OR Y > 28
      OR CS = 0 THEN  HOME : GOTO 105
880 I =  INT ((X - 14) / 2):J =  INT ((Y - 1
     4) / 2)
884  IF PT(I,J) <  > 1 THEN 860
886  COLOR= 14: PLOT X,Y
888  FOR A = 1 TO 3: PRINT  CHR$ (7): NEXT A
890  GOTO 866
900  TEXT : HOME : HTAB 8: PRINT "FIN DU JEU
     "
902  PRINT D$;"PR# 4": PRINT  CHR$ (0)
904  PRINT D$;"PR# 0"
905  PRINT D$;"IN# 0"
910  END
```

Programme LASER.BAS

```
5  REM
10  REM  LANCEMENT DU JEU LASER.MOUSE
15  REM
20  TEXT : HOME :D$ =  CHR$ (4)
30  VTAB 6: HTAB 15: INVERSE : PRINT "LASER.
     MOUSE": NORMAL : PRINT : PRINT
40  PRINT "Pour pouvoir jouer, vous devez av
     oir:": PRINT
45  PRINT " - Un Apple //c ou //e avec 65C02
     "
50  PRINT " - Une souris connectée en Slot 4
     (//e)"
52  PRINT " - Un moniteur ou TV couleur"
55  PRINT : PRINT "Si vous ne remplissez pas
     ces conditions"
60  PRINT "frappez ESCape ou CTL-C, si OK RE
     TURN": PRINT : HTAB 15: PRINT "==> ";:
     GET A$
70  IF A$ =  CHR$ (27) OR A$ =  CHR$ (3) THE
     N 100
80  PRINT : VTAB 8: CALL  - 958: PRINT "Char
     ge LASER.SBR"
82  PRINT D$"BLOAD LASER.SBR,A$7000"
85  PRINT "Charge et lance LASER.MOUSE"
90  PRINT D$"BRUN LASER.MOUSE,A$6000"
95  CALL 24576
100  HOME
```

Désassemblage Récapitulation LASER.MOUSE

Après avoir saisi ces trois colonnes de codes sous moniteur, vous les sauvegarderez par :

BSAVE LASER.MOUSE,A$6000,L$343

↓ ↓ ↓

```
6000: A9 76      LDA #76
6002: 85 1A      STA 1A
6004: 64 19      STZ 19
6006: EA         NOP
6007: EA         NOP
6008: EA         NOP
6009: EA         NOP
600A: A9 C4      LDA #C4
600C: 85 07      STA 07
600E: 64 06      STZ 06
6010: A0 19      LDY #19
6012: B1 06      LDA (06),Y
6014: AA         TAX
6015: 20 00 70   JSR 7000
6018: A9 24      LDA #24
601A: 8D F8 04   STA 04F8
601D: 9C F8 05   STZ 05F8
6020: 9C 78 04   STZ 0478
6023: 9C 78 05   STZ 0578
6026: A0 17      LDY #17
6028: B1 06      LDA (06),Y
602A: AA         TAX
602B: DA         PHX
602C: A9 00      LDA #00
602E: 20 00 70   JSR 7000
6031: FA         PLX
6032: A9 01      LDA #01
6034: 20 00 70   JSR 7000
6037: A0 12      LDY #12
6039: B1 06      LDA (06),Y
603B: AA         TAX
603C: A9 01      LDA #01
603E: 20 00 70   JSR 7000
6041: A0 15      LDY #15
6043: B1 06      LDA (06),Y
6045: AA         TAX
6046: 20 00 70   JSR 7000
6049: 20 0B 70   JSR 700B
604C: 20 38 71   JSR 7138
604F: EA         NOP
6050: A0 14      LDY #14
6052: B1 06      LDA (06),Y
6054: AA         TAX
6055: 20 00 70   JSR 7000
6058: E6 4E      INC 4E
605A: AC 7C 04   LDY 047C
605D: AD FC 04   LDA 04FC
6060: AA         TAX
6061: 20 71 F8   JSR F871
6064: 85 1F      STA 1F
6066: A5 1E      LDA 1E
6068: 20 64 F8   JSR F864
606B: 8A         TXA
606C: 20 00 F8   JSR F800
606F: 20 A8 FC   JSR FCA8
6072: A5 1F      LDA 1F
6074: 20 64 F8   JSR F864
6077: 8A         TXA
6078: 20 00 F8   JSR F800
```

```
607B: 2C 7C 07    BIT 077C
607E: 10 D0       BPL 6050 {-30}
6080: E6 4F       INC 4F
6082: AE 7C 04    LDX 047C
6085: E0 09       CPX #09
6087: 30 0D       BMI 6096 {+0D}
6089: A5 1D       LDA 1D
608B: 29 07       AND #07
608D: D0 5C       BNE 60EB {+5C}
608F: EA          NOP
6090: 20 E0 71    JSR 71E0
6093: 80 BB       BRA 6050 {-45}
6095: EA          NOP
6096: AC FC 04    LDY 04FC
6099: C0 1A       CPY #1A
609B: 30 03       BMI 60A0 {+03}
609D: 4C 30 63    JMP 6330
60A0: A5 1D       LDA 1D
60A2: D0 27       BNE 60CB {+27}
60A4: C0 0A       CPY #0A
60A6: 10 0C       BPL 60B4 {+0C}
60A8: A9 0B       LDA #0B
60AA: 85 1E       STA 1E
60AC: 85 30       STA 30
60AE: A9 01       LDA #01
60B0: 85 1D       STA 1D
60B2: 80 11       BRA 60C5 {+11}
60B4: C0 13       CPY #13
60B6: 30 09       BMI 60C1 {+09}
60B8: A9 05       LDA #05
60BA: 85 1D       STA 1D
60BC: 20 F0 71    JSR 71F0
60BF: 80 04       BRA 60C5 {+04}
60C1: A9 03       LDA #03
60C3: 85 1D       STA 1D
60C5: 20 A0 71    JSR 71A0
60C8: 4C 50 60    JMP 6050
60CB: C0 0A       CPY #0A
60CD: 10 03       BPL 60D2 {+03}
60CF: 4C 41 60    JMP 6041
60D2: C0 13       CPY #13
60D4: 30 06       BMI 60DC {+06}
60D6: 20 F0 71    JSR 71F0
60D9: 20 D3 71    JSR 71D3
60DC: A5 1E       LDA 1E
60DE: D0 07       BNE 60E7 {+07}
60E0: 20 E0 71    JSR 71E0
60E3: 64 1E       STZ 1E
60E5: 64 1D       STZ 1D
60E7: 4C 50 60    JMP 6050
60EA: EA          NOP
60EB: 85 1D       STA 1D
60ED: A5 1F       LDA 1F
60EF: F0 F6       BEQ 60E7 {-0A}
60F1: E0 0A       CPX #0A
60F3: 30 F2       BMI 60E7 {-0E}
60F5: E0 21       CPX #21
60F7: 10 EE       BPL 60E7 {-12}
60F9: AC FC 04    LDY 04FC
60FC: C0 0A       CPY #0A
60FE: 30 E7       BMI 60E7 {-19}
6100: C0 21       CPY #21
6102: 10 E3       BPL 60E7 {-1D}
6104: EA          NOP
6105: A5 1E       LDA 1E
6107: 20 64 F8    JSR F864
610A: 86 FE       STX FE
610C: 84 FF       STY FF
610E: A5 1D       LDA 1D
6110: E0 0E       CPX #0E
6112: 30 06       BMI 611A {+06}
6114: E0 1D       CPX #1D
6116: 30 07       BMI 611F {+07}
6118: 09 80       ORA #80
611A: 85 1D       STA 1D
611C: 4C 5A 61    JMP 615A
611F: C0 0E       CPY #0E
6121: 30 06       BMI 6129 {+06}
6123: C0 1D       CPY #1D
6125: 30 07       BMI 612E {+07}
6127: 09 80       ORA #80
6129: 85 1D       STA 1D
612B: 4C 20 62    JMP 6220
612E: 98          TYA
612F: 20 80 71    JSR 7180
6132: 85 1C       STA 1C
6134: A8          TAY
6135: 8A          TXA
6136: 20 80 71    JSR 7180
6139: 85 1B       STA 1B
613B: AA          TAX
613C: F0 03       BEQ 6141 {+03}
613E: 20 90 71    JSR 7190
6141: 85 19       STA 19
6143: B1 19       LDA (19),Y
6145: F0 0E       BEQ 6155 {+0E}
6147: A9 0E       LDA #0E
6149: 85 30       STA 30
614B: A4 FE       LDY FE
614D: A5 FF       LDA FF
614F: 20 00 F8    JSR F800
6152: 20 DD FB    JSR FBDD
6155: 4C 50 60    JMP 6050
6158: 00 00       BRK 00
615A: 20 28 72    JSR 7228
615D: A5 FF       LDA FF
615F: 20 80 71    JSR 7180
6162: 85 1C       STA 1C
6164: A8          TAY
6165: B1 19       LDA (19),Y
6167: F0 03       BEQ 616C {+03}
6169: 4C 00 63    JMP 6300
616C: 98          TYA
616D: F0 0D       BEQ 617C {+0D}
616F: 88          DEY
6170: B1 19       LDA (19),Y
6172: F0 03       BEQ 6177 {+03}
6174: 4C FB 61    JMP 61FB
6177: C8          INY
6178: C0 07       CPY #07
617A: F0 08       BEQ 6184 {+08}
617C: C8          INY
617D: B1 19       LDA (19),Y
617F: F0 03       BEQ 6184 {+03}
6181: 4C FB 61    JMP 61FB
6184: 18          CLC
6185: A6 1B       LDX 1B
6187: A4 1C       LDY 1C
6189: 24 1D       BIT 1D
618B: 30 08       BMI 6195 {+08}
618D: E8          INX
618E: E0 08       CPX #08
6190: 30 06       BMI 6198 {+06}
6192: 4C 06 62    JMP 6206
6195: CA          DEX
6196: 30 6E       BMI 6206 {+6E}
6198: 8A          TXA
6199: F0 03       BEQ 619E {+03}
619B: 20 90 71    JSR 7190
619E: 85 19       STA 19
61A0: B1 19       LDA (19),Y
61A2: F0 03       BEQ 61A7 {+03}
61A4: 4C 00 63    JMP 6300
61A7: 18          CLC
61A8: 08          PHP
61A9: 98          TYA
61AA: F0 0C       BEQ 61B8 {+0C}
61AC: 88          DEY
61AD: B1 19       LDA (19),Y
61AF: F0 02       BEQ 61B3 {+02}
61B1: 38          SEC
61B2: 08          PHP
61B3: C8          INY
61B4: C0 07       CPY #07
61B6: F0 14       BEQ 61CC {+14}
61B8: C8          INY
61B9: B1 19       LDA (19),Y
61BB: F0 0F       BEQ 61CC {+0F}
61BD: 28          PLP
61BE: 90 03       BCC 61C3 {+03}
61C0: 4C FB 61    JMP 61FB
61C3: A5 1D       LDA 1D
61C5: 09 80       ORA #80
61C7: 85 1D       STA 1D
61C9: 4C 60 62    JMP 6260
61CC: 28          PLP
61CD: 90 09       BCC 61D8 {+09}
61CF: A5 1D       LDA 1D
61D1: 29 0F       AND #0F
61D3: 85 1D       STA 1D
61D5: 4C 60 62    JMP 6260
61D8: E6 4E       INC 4E
61DA: A5 1D       LDA 1D
61DC: 29 06       AND #06
61DE: F0 03       BEQ 61E3 {+03}
61E0: 20 92 72    JSR 7292
61E3: 24 1D       BIT 1D
61E5: 10 08       BPL 61EF {+08}
61E7: A5 1B       LDA 1B
61E9: F0 1B       BEQ 6206 {+1B}
61EB: C6 1B       DEC 1B
61ED: 80 08       BRA 61F7 {+08}
61EF: A5 1B       LDA 1B
61F1: C9 07       CMP #07
61F3: F0 11       BEQ 6206 {+11}
61F5: E6 1B       INC 1B
61F7: 4C 84 61    JMP 6184
61FA: EA          NOP
61FB: 20 C0 72    JSR 72C0
61FE: A4 FE       LDY FE
6200: 20 46 72    JSR 7246
6203: 4C 00 63    JMP 6300
6206: A5 1C       LDA 1C
6208: 20 88 71    JSR 7188
620B: 85 FF       STA FF
620D: 20 C0 72    JSR 72C0
6210: 20 28 72    JSR 7228
6213: 4C 00 63    JMP 6300
6216: EA          NOP
6217: EA          NOP
6218: EA          NOP
6219: EA          NOP
621A: EA          NOP
621B: EA          NOP
621C: EA          NOP
621D: EA          NOP
621E: EA          NOP
621F: EA          NOP
6220: 20 60 72    JSR 7260
6223: A5 FE       LDA FE
6225: 20 80 71    JSR 7180
6228: AA          TAX
6229: 85 1B       STA 1B
622B: F0 03       BEQ 6230 {+03}
622D: 20 90 71    JSR 7190
6230: 85 19       STA 19
6232: A4 1C       LDY 1C
6234: B1 19       LDA (19),Y
6236: F0 03       BEQ 623B {+03}
6238: 4C 00 63    JMP 6300
623B: 8A          TXA
623C: F0 15       BEQ 6253 {+15}
623E: CA          DEX
623F: 8A          TXA
6240: F0 03       BEQ 6245 {+03}
6242: 20 90 71    JSR 7190
6245: 85 19       STA 19
6247: B1 19       LDA (19),Y
```

```
6249: F0 03        BEQ 624E {+03}
624B: 4C E4 62     JMP 62E4
624E: E8           INX
624F: E0 07        CPX #07
6251: F0 0D        BEQ 6260 {+0D}
6253: E8           INX
6254: 20 90 71     JSR 7190
6257: 85 19        STA 19
6259: B1 19        LDA (19),Y
625B: F0 03        BEQ 6260 {+03}
625D: 4C E4 62     JMP 62E4
6260: 18           CLC
6261: A4 1C        LDY 1C
6263: A6 1B        LDX 1B
6265: 8A           TXA
6266: F0 03        BEQ 626B {+03}
6268: 20 90 71     JSR 7190
626B: 85 19        STA 19
626D: 24 1D        BIT 1D
626F: 30 08        BMI 6279 {+08}
6271: C8           INY
6272: C0 08        CPY #08
6274: 30 06        BMI 627C {+06}
6276: 4C EF 62     JMP 62EF
6279: 88           DEY
627A: 30 73        BMI 62EF {+73}
627C: B1 19        LDA (19),Y
627E: F0 03        BEQ 6283 {+03}
6280: 4C 00 63     JMP 6300
6283: 18           CLC
6284: 08           PHP
6285: 8A           TXA
6286: F0 14        BEQ 629C {+14}
6288: CA           DEX
6289: 8A           TXA
628A: F0 03        BEQ 628F {+03}
628C: 20 90 71     JSR 7190
628F: 85 19        STA 19
6291: B1 19        LDA (19),Y
6293: F0 02        BEQ 6297 {+02}
6295: 38           SEC
6296: 08           PHP
6297: E8           INX
6298: E0 07        CPX #07
629A: F0 19        BEQ 62B5 {+19}
629C: E8           INX
629D: 20 90 71     JSR 7190
62A0: 85 19        STA 19
62A2: B1 19        LDA (19),Y
62A4: F0 0F        BEQ 62B5 {+0F}
62A6: 28           PLP
62A7: 90 03        BCC 62AC {+03}
62A9: 4C E4 62     JMP 62E4
62AC: A5 1D        LDA 1D
62AE: 09 80        ORA #80
62B0: 85 1D        STA 1D
62B2: 4C 84 61     JMP 6184
62B5: 28           PLP
62B6: 90 09        BCC 62C1 {+09}
62B8: A5 1D        LDA 1D
62BA: 29 0F        AND #0F
62BC: 85 1D        STA 1D
62BE: 4C 84 61     JMP 6184
62C1: E6 4F        INC 4F
62C3: A5 1D        LDA 1D
62C5: 29 06        AND #06
62C7: F0 03        BEQ 62CC {+03}
62C9: 20 92 72     JSR 7292
62CC: 24 1D        BIT 1D
62CE: 10 08        BPL 62D8 {+08}
62D0: A5 1C        LDA 1C
62D2: F0 18        BEQ 62EC {+18}
62D4: C6 1C        DEC 1C
62D6: 80 08        BRA 62E0 {+08}
62D8: A5 1C        LDA 1C
62DA: C9 07        CMP #07
62DC: F0 0E        BEQ 62EC {+0E}
62DE: E6 1C        INC 1C
62E0: 4C 60 62     JMP 6260
62E3: EA           NOP
62E4: 20 C0 72     JSR 72C0
62E7: A6 FF        LDX FF
62E9: 20 78 72     JSR 7278
62EC: 4C 00 63     JMP 6300
62EF: 20 C0 72     JSR 72C0
62F2: A5 1B        LDA 1B
62F4: 20 88 71     JSR 7188
62F7: 85 FE        STA FE
62F9: 20 60 72     JSR 7260
62FC: EA           NOP
62FD: EA           NOP
62FE: EA           NOP
62FF: EA           NOP
6300: A5 1D        LDA 1D
6302: 29 0F        AND #0F
6304: 85 1D        STA 1D
6306: C6 1E        DEC 1E
6308: F0 0A        BEQ 6314 {+0A}
630A: A5 1D        LDA 1D
630C: 29 06        AND #06
630E: D0 07        BNE 6317 {+07}
6310: C6 1E        DEC 1E
6312: 80 03        BRA 6317 {+03}
6314: 20 E0 71     JSR 71E0
6317: A5 1E        LDA 1E
6319: 20 64 F8     JSR F864
631C: A9 80        LDA #80
631E: 20 A8 FC     JSR FCA8
6321: 4C 50 60     JMP 6050
6324: EA           NOP
6325: EA           NOP
6326: EA           NOP
6327: EA           NOP
6328: EA           NOP
6329: EA           NOP
632A: EA           NOP
632B: EA           NOP
632C: EA           NOP
632D: EA           NOP
632E: EA           NOP
632F: EA           NOP
6330: A0 12        LDY #12
6332: B1 06        LDA (06),Y
6334: AA           TAX
6335: A9 00        LDA #00
6337: 20 00 70     JSR 7000
633A: 20 39 FB     JSR FB39
633D: 20 58 FC     JSR FC58
6340: 4C D0 03     JMP 03D0
```

Désassemblage Récapitulation LASER.SBR

Après avoir saisi ces trois colonnes de codes sous moniteur, vous les sauvegarderez par :

BSAVE LASER.SBR,A$7000,L$393

↓ ↓ ↓

```
7000: 8E 08 70     STX 7008
7003: A2 C4        LDX #C4
7005: A0 40        LDY #40
7007: 20 3D C4     JSR C43D
700A: 60           RTS
700B: 20 40 FB     JSR FB40
700E: 20 36 F8     JSR F836
7011: A9 0D        LDA #0D
7013: 20 64 F8     JSR F864
7016: A0 02        LDY #02
7018: A9 21        LDA #21
701A: 85 2D        STA 2D
701C: A9 03        LDA #03
701E: 20 28 F8     JSR F828
7021: A0 08        LDY #08
7023: 84 2C        STY 2C
7025: A9 03        LDA #03
7027: 20 28 F8     JSR F828
702A: A0 02        LDY #02
702C: 98           TYA
702D: 20 19 F8     JSR F819
7030: A0 02        LDY #02
7032: A9 22        LDA #22
7034: 20 19 F8     JSR F819
7037: A9 07        LDA #07
7039: 85 2C        STA 2C
703B: A0 03        LDY #03
703D: A9 0A        LDA #0A
703F: 20 19 F8     JSR F819
7042: A0 03        LDY #03
7044: A9 12        LDA #12
7046: 20 19 F8     JSR F819
7049: A0 03        LDY #03
704B: A9 1A        LDA #1A
704D: 20 19 F8     JSR F819
7050: A9 0B        LDA #0B
7052: 20 64 F8     JSR F864
7055: A9 08        LDA #08
7057: 85 2D        STA 2D
7059: A9 04        LDA #04
705B: A0 06        LDY #06
705D: 20 28 F8     JSR F828
7060: A9 05        LDA #05
7062: 85 2C        STA 2C
7064: A0 04        LDY #04
7066: A9 08        LDA #08
7068: 20 19 F8     JSR F819
706B: A0 04        LDY #04
706D: A9 07        LDA #07
706F: 20 00 F8     JSR F800
7072: A9 0C        LDA #0C
7074: 20 64 F8     JSR F864
7077: A9 10        LDA #10
7079: 85 2D        STA 2D
707B: A0 04        LDY #04
707D: A9 0C        LDA #0C
707F: 20 28 F8     JSR F828
7082: A9 06        LDA #06
7084: 85 2C        STA 2C
7086: A0 05        LDY #05
7088: A9 0C        LDA #0C
708A: 20 19 F8     JSR F819
708D: A0 05        LDY #05
708F: A9 0E        LDA #0E
7091: 20 00 F8     JSR F800
7094: A9 01        LDA #01
7096: 20 64 F8     JSR F864
7099: A9 17        LDA #17
709B: 85 2D        STA 2D
709D: A0 06        LDY #06
709F: 84 2C        STY 2C
70A1: A9 15        LDA #15
70A3: 20 28 F8     JSR F828
70A6: A0 03        LDY #03
70A8: A9 14        LDA #14
70AA: 20 19 F8     JSR F819
70AD: A0 03        LDY #03
70AF: A9 18        LDA #18
```

```
70B1: 20 19 F8    JSR F819
70B4: A0 04       LDY #04
70B6: A9 15       LDA #15
70B8: 20 28 F8    JSR F828
70BB: A9 0F       LDA #0F
70BD: 20 64 F8    JSR F864
70C0: A2 1C       LDX #1C
70C2: 8A          TXA
70C3: A0 04       LDY #04
70C5: 20 19 F8    JSR F819
70C8: E8          INX
70C9: E8          INX
70CA: E0 21       CPX #21
70CC: 30 F4       BMI 70C2 {-0C}
70CE: A9 1D       LDA #1D
70D0: A0 04       LDY #04
70D2: 20 00 F8    JSR F800
70D5: A9 1F       LDA #1F
70D7: A0 06       LDY #06
70D9: 20 00 F8    JSR F800
70DC: A9 0D       LDA #0D
70DE: 20 64 F8    JSR F864
70E1: 38          SEC
70E2: A9 0A       LDA #0A
70E4: 85 FF       STA FF
70E6: 69 01       ADC #01
70E8: 85 FC       STA FC
70EA: 69 02       ADC #02
70EC: 85 FE       STA FE
70EE: 69 0E       ADC #0E
70F0: 85 FB       STA FB
70F2: 69 02       ADC #02
70F4: 85 FD       STA FD
70F6: 69 02       ADC #02
70F8: 85 FA       STA FA
70FA: A2 03       LDX #03
70FC: A0 0E       LDY #0E
70FE: B5 F9       LDA F9,X
7100: 85 2C       STA 2C
7102: 85 2D       STA 2D
7104: B5 FC       LDA FC,X
7106: 20 28 F8    JSR F828
7109: 5A          PHY
710A: 98          TYA
710B: B4 FC       LDY FC,X
710D: 20 19 F8    JSR F819
7110: 7A          PLY
7111: C8          INY
7112: C8          INY
7113: C0 1D       CPY #1D
7115: 30 ED       BMI 7104 {-13}
7117: CA          DEX
7118: D0 E2       BNE 70FC {-1E}
711A: A9 0C       LDA #0C
711C: 85 1E       STA 1E
711E: 64 1D       STZ 1D
7120: A9 D0       LDA #D0
7122: 85 08       STA 08
7124: A9 72       LDA #72
7126: 85 09       STA 09
7128: 20 58 FC    JSR FC58
712B: A0 00       LDY #00
712D: B1 08       LDA (08),Y
712F: F0 06       BEQ 7137 {+06}
7131: 20 ED FD    JSR FDED
7134: C8          INY
7135: 80 F6       BRA 712D {-0A}
7137: 60          RTS
7138: A9 00       LDA #00
713A: 85 19       STA 19
713C: A0 41       LDY #41
713E: 88          DEY
713F: 91 19       STA (19),Y
7141: 10 FB       BPL 713E {-05}
7143: A2 02       LDX #02
7145: A5 4E       LDA 4E
7147: 48          PHA
7148: 29 07       AND #07
714A: 0A          ASL
714B: 0A          ASL
714C: 0A          ASL
714D: 85 19       STA 19
714F: 68          PLA
7150: 4A          LSR
7151: 4A          LSR
7152: 4A          LSR
7153: 29 07       AND #07
7155: A8          TAY
7156: A9 01       LDA #01
7158: 91 19       STA (19),Y
715A: CA          DEX
715B: F0 04       BEQ 7161 {+04}
715D: A5 4F       LDA 4F
715F: 80 E6       BRA 7147 {-1A}
7161: A5 4E       LDA 4E
7163: 4A          LSR
7164: 4A          LSR
7165: 29 38       AND #38
7167: 85 19       STA 19
7169: A5 4F       LDA 4F
716B: 4A          LSR
716C: 4A          LSR
716D: 4A          LSR
716E: 4A          LSR
716F: 4A          LSR
7170: 29 07       AND #07
7172: A8          TAY
7173: B1 19       LDA (19),Y
7175: F0 03       BEQ 717A {+03}
7177: C8          INY
7178: 80 F9       BRA 7173 {-07}
717A: A9 01       LDA #01
717C: 91 19       STA (19),Y
717E: 60          RTS
717F: EA          NOP
7180: 38          SEC
7181: E9 0E       SBC #0E
7183: 4A          LSR
7184: 18          CLC
7185: 60          RTS
7186: EA          NOP
7187: EA          NOP
7188: 0A          ASL
7189: 18          CLC
718A: 69 0E       ADC #0E
718C: 60          RTS
718D: 00 00       BRK 00
718F: 00 DA       BRK DA
7191: A9 00       LDA #00
7193: 18          CLC
7194: 69 08       ADC #08
7196: CA          DEX
7197: D0 FA       BNE 7193 {-06}
7199: FA          PLX
719A: 60          RTS
719B: 00 00       BRK 00
719D: 00 00       BRK 00
719F: 00 A9       BRK A9
71A1: 00 20       BRK 20
71A3: 64 F8       STZ F8
71A5: A9 10       LDA #10
71A7: 85 2D       STA 2D
71A9: A0 05       LDY #05
71AB: A9 0C       LDA #0C
71AD: 20 28 F8    JSR F828
71B0: C8          INY
71B1: C0 07       CPY #07
71B3: 30 F6       BMI 71AB {-0A}
71B5: A5 1E       LDA 1E
71B7: 20 64 F8    JSR F864
71BA: A0 04       LDY #04
71BC: A9 0C       LDA #0C
71BE: 20 28 F8    JSR F828
71C1: A9 06       LDA #06
71C3: 85 2C       STA 2C
71C5: A9 0C       LDA #0C
71C7: A0 05       LDY #05
71C9: 20 19 F8    JSR F819
71CC: A0 05       LDY #05
71CE: A9 10       LDA #10
71D0: 20 19 F8    JSR F819
71D3: A9 18       LDA #18
71D5: 85 08       STA 08
71D7: A9 73       LDA #73
71D9: 20 26 71    JSR 7126
71DC: 60          RTS
71DD: EA          NOP
71DE: EA          NOP
71DF: EA          NOP
71E0: A9 60       LDA #60
71E2: 85 08       STA 08
71E4: A9 73       LDA #73
71E6: 20 26 71    JSR 7126
71E9: 60          RTS
71EA: EA          NOP
71EB: EA          NOP
71EC: EA          NOP
71ED: EA          NOP
71EE: EA          NOP
71EF: EA          NOP
71F0: A9 01       LDA #01
71F2: 85 30       STA 30
71F4: A9 38       LDA #38
71F6: 85 19       STA 19
71F8: A9 07       LDA #07
71FA: 85 1B       STA 1B
71FC: A0 07       LDY #07
71FE: B1 19       LDA (19),Y
7200: F0 12       BEQ 7214 {+12}
7202: 84 1C       STY 1C
7204: A5 1B       LDA 1B
7206: 20 88 71    JSR 7188
7209: A8          TAY
720A: A5 1C       LDA 1C
720C: 20 88 71    JSR 7188
720F: 20 00 F8    JSR F800
7212: A4 1C       LDY 1C
7214: 88          DEY
7215: 10 E7       BPL 71FE {-19}
7217: C6 1B       DEC 1B
7219: 30 09       BMI 7224 {+09}
721B: A5 19       LDA 19
721D: 38          SEC
721E: E9 08       SBC #08
7220: 85 19       STA 19
7222: 80 D8       BRA 71FC {-28}
7224: 60          RTS
7225: EA          NOP
7226: EA          NOP
7227: EA          NOP
7228: 64 1B       STZ 1B
722A: 64 19       STZ 19
722C: A0 0A       LDY #0A
722E: A9 C8       LDA #C8
7230: 8D 53 72    STA 7253
7233: 24 1D       BIT 1D
7235: 10 0F       BPL 7246 {+0F}
7237: A0 20       LDY #20
7239: A9 07       LDA #07
723B: 85 1B       STA 1B
723D: A9 38       LDA #38
723F: 85 19       STA 19
7241: A9 88       LDA #88
7243: 8D 53 72    STA 7253
7246: 84 FE       STY FE
7248: A5 FF       LDA FF
724A: 20 71 F8    JSR F871
724D: F0 07       BEQ 7256 {+07}
```

```
724F: C9 0D        CMP #0D
7251: F0 03        BEQ 7256 (+03)
7253: C8           INY
7254: 80 F0        BRA 7246 (-10)
7256: A5 FF        LDA FF
7258: 20 00 F8     JSR F800
725B: 20 DD FB     JSR FBDD
725E: 60           RTS
725F: EA           NOP
7260: 64 1C        STZ 1C
7262: A2 0A        LDX #0A
7264: A9 E8        LDA #E8
7266: 8D 84 72     STA 7284
7269: 24 1D        BIT 1D
726B: 10 0B        BPL 7278 (+0B)
726D: A2 20        LDX #20
726F: A9 07        LDA #07
7271: 85 1C        STA 1C
7273: A9 CA        LDA #CA
7275: 8D 84 72     STA 7284
7278: A4 FE        LDY FE
727A: 8A           TXA
727B: 20 71 F8     JSR F871
727E: F0 07        BEQ 7287 (+07)
7280: C9 0D        CMP #0D
7282: F0 03        BEQ 7287 (+03)
7284: CA           DEX
7285: 80 F3        BRA 727A (-0D)
7287: 86 FF        STX FF
7289: 8A           TXA
728A: 20 00 F8     JSR F800
728D: 20 DD FB     JSR FBDD
7290: 60           RTS
7291: EA           NOP
7292: A5 1B        LDA 1B
7294: 20 88 71     JSR 7188
7297: A8           TAY
7298: A5 1C        LDA 1C
729A: 20 88 71     JSR 7188
729D: AA           TAX
729E: 20 71 F8     JSR F871
72A1: 85 1F        STA 1F
72A3: A5 1E        LDA 1E
72A5: 85 30        STA 30
72A7: 8A           TXA
72A8: 20 00 F8     JSR F800
72AB: 20 A8 FC     JSR FCA8
72AE: A5 1F        LDA 1F
72B0: 20 64 F8     JSR F864
72B3: 8A           TXA
72B4: 20 00 F8     JSR F800
72B7: A5 1E        LDA 1E
72B9: 20 64 F8     JSR F864
72BC: 60           RTS
72BD: EA           NOP
72BE: EA           NOP
72BF: EA           NOP
72C0: A5 1D        LDA 1D
72C2: 49 80        EOR #80
72C4: 85 1D        STA 1D
72C6: 60           RTS
72C7: EA           NOP
72C8: EA           NOP
72C9: E5 F5        SBC F5
72CB: A0 CE        LDY #CE
72CD: CF D2 CD 8D  CMP 8DCDD2
72D1: A0 A0        LDY #A0
72D3: 0A           ASL
72D4: A0 BD        LDY #BD
72D6: BE A0 CA     LDX CAA0,Y
72D9: E5 F5        SBC F5
72DB: A0 CE        LDY #CE
72DD: CF D2 CD C1  CMP C1CDD2
72E1: CC A0 A0     CPY A0A0
72E4: A0 06        LDY #06
72E6: A0 BD        LDY #BD
72E8: BE A0 CA     LDX CAA0,Y
72EB: E5 F5        SBC F5
72ED: A0 C6        LDY #C6
72EF: C1 C3        CMP (C3,X)
72F1: C9 CC        CMP #CC
72F3: C5 8D        CMP 8D
72F5: A0 A0        LDY #A0
72F7: 04 A0        TSB A0
72F9: BD BE A0     LDA A0BE,X
72FC: C4 C5        CPY C5
72FE: CD CF EE     CMP EECF
7301: F3 F4        SBC (F4,S),Y
7303: F2 E1        SBC (E1)
7305: F4 E9 EF     PEA EFE9
7308: EE A0 A0     INC A0A0
730B: A0 13        LDY #13
730D: A0 BD        LDY #BD
730F: BE A0 D3     LDX D3A0,Y
7312: D4 CF        PEI CF
7314: D0 00        BNE 7316 (+00)
7316: 00 00        BRK 00
7318: 8D A0 A0     STA A0A0
731B: 0A           ASL
731C: A0 BD        LDY #BD
731E: BE A0 D2     LDX D2A0,Y
7321: E5 CA        SBC CA
7323: CF D5 C5 D2  CMP D2C5D5
7327: A0 A0        LDY #A0
7329: A0 03        LDY #03
732B: A0 BD        LDY #BD
732D: BE A0 C3     LDX C3A0,Y
7330: CF CE D4 C9  CMP C9D4CE
7334: CE D5 C5     DEC C5D5
7337: D2 8D        CMP (8D)
7339: A0 A0        LDY #A0
733B: 04 A0        TSB A0
733D: BD BE A0     LDA A0BE,X
7340: C4 EF        CPY EF
7342: EE EE E5     INC E5EE
7345: A0 D3        LDY #D3
7347: CF CC D5 D4  CMP D4D5CC
734B: C9 CF        CMP #CF
734D: CE A0 A0     DEC A0A0
7350: A0 13        LDY #13
7352: A0 BD        LDY #BD
7354: BE A0 D3     LDX D3A0,Y
7357: D4 CF        PEI CF
7359: D0 00        BNE 735B (+00)
735B: 00 00        BRK 00
735D: 00 00        BRK 00
735F: 00 8D        BRK 8D
7361: A0 A0        LDY #A0
7363: CD EF E4     CMP E4EF
7366: E5 A0        SBC A0
7368: EE EF EE     INC EEEF
736B: A0 F3        LDY #F3
736D: E5 EC        SBC EC
736F: E5 E3        SBC E3
7371: F4 E9 EF     PEA EFE9
7374: EE EE E5     INC E5EE
7377: 8D A0 A0     STA A0A0
737A: EF F5 A0 F0  SBC F0A0F5
737E: EC F5 F3     CPX F3F5
7381: A0 E4        LDY #E4
7383: E5 A0        SBC A0
7385: E3 EF        SBC EF,S
7387: F5 EC        SBC EC,X
7389: E5 F5        SBC F5
738B: F2 00        SBC (00)
738D: 00 00        BRK 00
738F: 00 00        BRK 00
7391: 00 00        BRK 00
```

BugByter

A. Avrane, dans le numéro 13 de Pom's, regrettait l'impossibilité d'imprimer depuis BugByter. Pour remédier à cet état de chose, voici un petit patch et la procédure de mise en place :

```
Booter sur la disquette Master DOS 3.3
CALL - 151
BLOAD BUGBYTER, A$1000
1729:4C 00 EA
1679:4C 20 EA EA
2A00: C9 9B D0 03 4C F8 D6 C9 D7 D0 06 8D 1D EA 4C
   2D D7 C9 C5 D0 05 A9 00 8D 1D EA 4C 2D D7 00 00
   EA A9 80 2C 1D EA 10 3B AE AC E7 CA BD 10 E9 8D
   4B EA BD 28 E9 8D 4C EA AC B7 E7 A9 50 8D 79 06
   A9 00 8D F9 05 85 36 A9 C1 85 37 B9 28 06 29 7F
   C9 20 B0 02 09 40 20 00 C1 C8 C0 27 D0 ED A9 0D
   20 00 C1 68 8D 98 E7 4C 7D D6 00
BSAVE BUGBYTER, A$1000, L$1A69 (sur une copie !)
```

En mode SINGLE STEP, l'appui sur W(rite) démarre l'impression et l'appui sur E(nd) l'arrête. Une fois la commande W donnée, on peut passer en mode Trace si on le désire.

Michel Duroc, 97220 Trinité

On trouve plus facilement avec 10 ans

Certaines évolutions sont aussi importantes que des révolutions.

Le plus difficile quand on débute, c'est de débuter. Car toutes les propositions d'emploi demandent une certaine expérience. Chercher un emploi est donc un problème insoluble. Pas pour l'Apple IIGS, il débute avec 10 ans d'expérience.

Comment? C'est très simple, il a une architecture double : un nouvel ordinateur puissant traitant le graphique et le son, entoure un Apple II classique et miniaturisé. Il profite ainsi de 10 ans de développement de logiciels et d'expérience. Un grand progrès réalisé en partie grâce au microprocesseur 65C816, un 16 bits descendant du 65C02 de l'Apple II. L'Apple IIGS fonctionne à deux vitesses : 2,8 MHz en mode natif; en mode émulation (c'est-à-dire lorsque vous utilisez un programme de votre ancien Apple II) vous avez le choix entre 1 MHz et 2,8 MHz.

Débuter connu, c'est 16.000 propositions d'emploi.

Avec l'IWM (Integrated Woz Machine*) l'Apple IIGS peut recevoir indifféremment des lecteurs de disquettes 3,5 pouces 800 Ko ou des lecteurs 5,25 pouces 140 Ko, ou les faire cohabiter, ce qui contribue à vous faire profiter de la plus grande bibliothèque de logiciels du monde. Le nombre de 16.000 logiciels étant estimatif car la seule chose qu'un Apple IIGS soit incapable de calculer, c'est le nombre d'emplois qu'il est capable de tenir.

Avec le nouveau système d'exploitation ProDOS, l'Apple IIGS permet la hiérarchisation des fichiers et des catalogues à la manière de

** En hommage à Steve Wozniak, co-fondateur de la Société Apple et créateur de l'Apple II.*

C.L.M. & BBDO.

un emploi quand on débute d'expérience.

Macintosh, et la connexion de nouveaux périphériques.

G et S, cela veut dire Graphique et son.

L'imagination débridée des développeurs suffira-t-elle pour utiliser les 4096 nuances de couleurs disponibles ? C'est probable. Ce qui est certain, c'est qu'une résolution de 640 points sur 200 lignes pour 4 couleurs, 320 points sur 200 lignes pour 16 couleurs va donner beaucoup de relief à certaines démonstrations. Si vous n'en croyez pas vos yeux, vous n'en croirez pas vos oreilles non plus. Le coprocesseur "son" choisi par Apple a déjà une brillante carrière derrière lui puisqu'il est employé par les plus grands synthétiseurs du marché. C'est l'ENSONIQ, capable de générer 16 voix, et il ne lui manque même pas la parole.

Les relations nécessaires pour réussir.

En 10 années, l'Apple II a eu tout le loisir de se faire d'excellentes relations de travail, imprimante, modem, disque dur, que l'Apple IIGS continuera à entretenir et développer ; celui-ci peut désormais être connecté au réseau AppleTalk et gérer la LaserWriter réservée jusqu'à présent à Macintosh. Il peut aussi recevoir le disque dur SCSI.

L'Apple IIGS possède 7 connecteurs d'extension permettant, à l'aide d'une multitude de cartes d'interfaces, de le relier à tous les types d'unités périphériques.

Acheter un Apple, c'est entrer dans le Club Apple pour échanger des informations, accéder au support technique par téléphone 7 jours sur 7, ou aux services télématiques du Club.

Apple présente l'Apple IIGS.

Apple, le logo Apple, Apple IIGS, ProDOS, Macintosh, Apple Talk, LaserWriter et GS Paint sont des marques déposées par Apple Computer Inc.

Freeware

Un billard à l'essai

À la rédaction de Pom's, les pauses — rares et studieuses (1) — sont souvent consacrées à la cinématique des sphères, entendez le billard sur Macintosh. L'objet de notre passion est le Billard Parlour, freeware de Crandall, Gillespie et Lew mis à votre disposition sur la disquette 'Mac I'. La qualité de ce programme nous incite à vous le présenter plus en détail.

Présentation

La copie d'écran ci-contre illustre la qualité de la présentation de ce jeu bien réaliste. Disposé au centre de l'écran, le tapis vert, au-dessus, le compteur de points et en haut, le boulier.

À l'aide de la souris, on fait apparaître la queue de billard et on lui donne l'angle voulu ; la force de l'impact est déterminée par la position de la souris. Tout cela est très instinctif. Il ne faut pas taper trop violemment : outre que le réalisme en souffre, les billes pourraient passer les unes au travers des autres sans que le programme ne gère un choc.

Comme bien des freewares, le mode d'emploi et le programme sont en anglais, mais ce jeu est si connu que cela ne devrait pas constituer un obstacle.

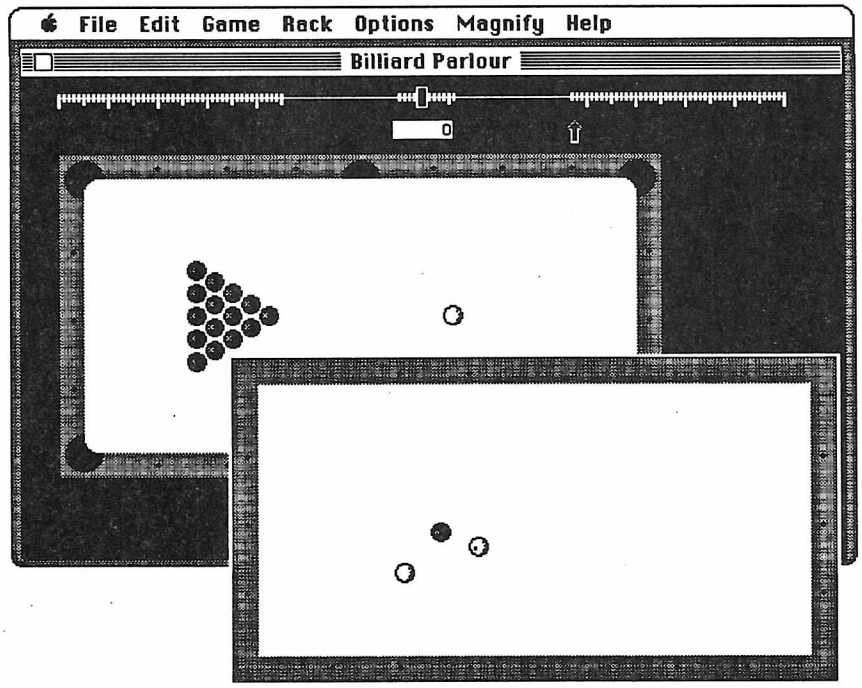
Les possibilités

Outre le jeu standard, il est possible de positionner les billes comme on le souhaite pour étudier un coup.

L'*effet* donné à la bille est matérialisé par un point sur un disque, point à positionner avant de jouer. À défaut, ce point est au zéro théorique, c'est-à-dire sans effet : le 'rétro' s'obtient en tapant plutôt au bas de la bille. De la même manière, on peut doser l'effet latéral.

Un bruit de collision qui rappelle fidèlement celui de l'ivoire est généré à chaque impact ; on peut le supprimer.

Le compteur évite toute ambiguïté sur l'obtention des points mais, en billard français, il ne fonctionne qu'en 'trois bandes'.



Possibilités qui n'existent pas dans les traditionnelles salles de billard : annuler et reprendre un coup, mais également rejouer un coup pour en contrôler les trajectoires ; un menu en propose le tracé.

Les billards

Ce programme est complet : il propose le jeu :

- avec les 3 billes du billard français ;
- avec 9 ou 15 billes numérotées (à jouer dans l'ordre) ;
- avec 15 billes numérotées et 6 non numérotées :
- avec 3, 6, 9, 10 ou 15 billes non numérotées.
- Lag, qui est un exercice de tir de précision.

Une loupe est disponible pour contrôler les numéros des billes numérotées mais ils sont en fait lisibles.

Le programme

Écrit en Rascal (2), il ne laisse pas de place au hasard : les coordonnées, les vitesses, l'analyse des collisions compte tenu de l'effet donné et les trajectoires font l'objet de calculs mathématiques (toujours sur des entiers 16 bits pour une question de rapidité). Le *help* donne, en anglais, des informations sur la méthode de calcul des collisions. L'affichage des billes est réalisé à l'aide de deux images pour donner une impression de rotation, mais il est un peu lent lorsque 21 billes sont sur le tapis.

(1) Précisions à l'attention du Rédacteur en Chef.

(2 Rascal de Metaresearh Inc. 1100 SE Woodward, Portland, OR 97202 États-Unis.

Essai Mac More 2

Philippe Mathieu

Nous supposerons dans cet article que le lecteur connaît déjà ThinkTank ou, pour le moins, les principes de fonctionnement des programmes de gestion d'idées dont ThinkTank a été le premier et brillant exemple.

Point fort : More et le Macintosh

More, digne successeur de ThinkTank, possède par rapport à son prédécesseur un fonctionnement plus "Macintosh", notamment grâce à son utilisation des ascenseurs. Il possède aussi une plus grande souplesse dans la réorganisation, à l'aide de la souris, de la hiérarchie des idées. Remarquons enfin la possibilité d'ouvrir simultanément plusieurs fenêtres et de Copier/Coller entre elles.

Point fort : présentation variée des documents

Outre les classiques liste et table des matières, More propose un dessin des titres et sous-titres sous la forme d'un organigramme, ainsi que la "projection" de planches et tableaux pour les conférences ou toute utilisation similaire.
Ces tableaux peuvent défiler, soit à intervalles réguliers, soit à la commande. Un défilement rapide permet de créer une sorte d'animation. Il est très aisée de coller dans More des dessins et graphiques créés par d'autres programmes, par exemple MacPaint ou MacDraw, et aussi par lui-même (organigramme collé comme dessin). Par ailleurs, More dispose de nombreuses options de polices caractères, styles, cadres et même de l'impression en couleurs sur l'ImageWriter II.

Point fort : grande puissance d'organisation

- possibilité de créer des "clones", des doubles d'une sélection qui seront modifiés si l'on modifie l'original (et inversement) ;
- "marquage", en liaison avec la recherche automatique, permettant de regrouper des titres ayant un certain contenu ;
- création et utilisation de modèles (gabarits) réutilisables dans divers fichiers.

Point fort : autres fonctionnalités utiles

More possède de larges facilités d'échange avec d'autres logiciels :
- accueil de documents provenant d'autres logiciels, y compris lui-même, avec possibilité d'insertion de ces documents dans le document courant ;
- envoi de documents vers d'autres logiciels ;
- gestion intéressante du temps : calendrier, dates, heures.

Le numérotage téléphonique automatique est possible, pour peu qu'on utilise un modem autorisant ce type d'opération.

Inconvénient : impression

Il est impossible de placer volontairement des sauts de page à l'impression, de sorte que les coupures se produisent de façon systématique quand la page est pleine (c'est-à-dire généralement au mauvais endroit).

Inconvénient : fenêtres de texte

Le traitement de texte reste assez rudimentaire. On ne peut par exemple pas changer les caractères, la taille ou le style d'un mot ou d'un groupe de mots à l'intérieur d'une même fenêtre. Heuresement, c'est possible d'une fenêtre à l'autre. On ne peut pas non plus voir à l'écran ce que sera l'apparence d'un texte après justification, celle-ci n'intervenant que lors de l'impression. Par ailleurs, les fenêtres de texte ne peuvent être ouvertes qu'une à une, après avoir sélectionné les sous-titres correspondants, ce qui est particulièrement fastidieux.

Quelques "gadgets" superflus

On peut se demander par ailleurs si certains gadgets, qui compliquent inutilement le maniement du logiciel, n'auraient pas pu être supprimés :
- trois modes de divisions en fenêtres ; on ne voit pas à quoi sert la division en diagonale ;
- trop souvent, trois méthodes pour obtenir le même résultat ; par exemple, pour la commande *Etendre*, qui peut être mise en œuvre par un menu déroulant, un clic ou une commande clavier.

Bien que mise en relief par l'auteur du programme, la possibilité d'associer des valeurs à des sous-titres et de les additioner automatiquement en remontant la hiérarchie des titres ne semble pas passionnante.

Attention, danger : les annulations

Il n'exite pas de commande d'annulation de l'opération précédente, comme c'est le cas maintenant dans pratiquement tous les programmes Macintosh. C'est une lacune.

Conclusion

More est un produit original et puissant, mais il serait bien agréable d'avoir rapidement une version More 2.1 dans laquelle la plupart des problèmes que nous avons indiqués ci-dessous seraient résolus.

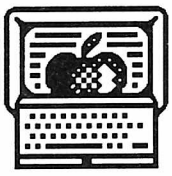
STRUCTUREXPERT ™

La solution pour programmer de manière structurée en MSBASIC ™ et ZBASIC ™. STRUCTUREXPERT ™ vous permet de disposer d'un ensemble d'analyses qui peuvent être visualisées (fenêtres "scrollables"), sauvées séparément sur fichiers, ou imprimées :

. Arborescence complète (descendante et ascendante) des appels GOSUB et CALL.
. Listes triées sur plusieurs types de critères des GOSUB et des CALL.
. Statistiques : fréquence d'appel de chaque routine.
. Détection d'erreurs logiques (boucles non contrôlées, récursivité illégale, labels et routines non utilisés).

STRUCTUREXPERT 495 F TTC + 20F de port
Paiements par CB ou chèques adressés à
Q.S.I. 149, rue OBERKAMPF 75011 PARIS
tel : (16)142493795

Pom's recherche *pour les mois de juillet et/ou août un ou deux stagiaires. Bonne maîtrise du Macintosh, du français et de l'orthographe requise.*
Envoyez votre CV à Hervé Thiriez, Éditions MEV, 12, rue d'Anjou 78000 Versailles avant le 15 juin.



T_Pom's

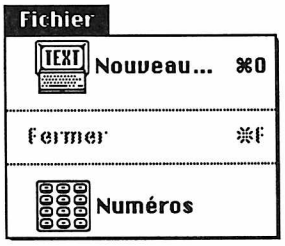
Jean-Luc Bazanegue

TPom's : Minitel, Macintosh & le 11

"TPom's", qui répond à une importante demande, autorise la récupération de l'annuaire électronique sous la forme d'un fichier 'texte' directement exploitable. Ce programme permet aussi, accessoirement, le traitement de fichiers créés à partir du programme "Minitel" (Pom's numéro 27). Il s'agit d'une application autonome écrite en assembleur MDS 68000, fonctionnant sur tous les Macintosh et ce quel que soit l'environnement. Comme pour les programmes "Minitel" et "InterPom's" (Pom's numéro 28) un câble-interface Minitel/ Macintosh est requis ; si vous ne l'avez pas déjà, vous pouvez le fabriquer en vous aidant des schémas publiés dans les numéros 27 ou 28 de Pom's ou, si vos dons ou votre temps ne vous le permettent pas, on peut aussi commander cette liaison indispensable directement à la revue. Pour cela, vous trouverez un bon de commande page 75.

Utilisation de TPom's

Le programme utilise quatre menus déroulants ; voici le premier :



Nouveau...

L'article 'Nouveau...' de ce menu sera employé pour choisir le nom du fichier destiné à recevoir les informations émises par TPom's. Les fichiers ainsi créés portent le type 'texte seul' MacWrite/DiskWrite (TEXT MACA), donc 'double cliquables' et exploitables avec n'importe quel logiciel capable de traiter ce format. Une fois le choix du nom effectué, il s'affiche dans la barre de titre de la fenêtre. Si le programme ne trouve pas de fichier ouvert, la barre de titre contient la mention '*Pas de fichier ouvert*'. L''équivalent clavier' de cet article est 'Commande-O'.

Fermer

Nous trouvons l'article 'Fermer' valide dans deux cas :

- un accessoire de bureau est affiché et actif. Dans ce cas, une action sur cet article – ou son équivalent clavier 'Commande-F' ferme l'accessoire en question ;
- un fichier est ouvert. Le traitement consiste alors en la fermeture du fichier courant, le remplacement du nom de fichier dans la barre de titre par la mention 'Pas de fichier ouvert', et le changement de la forme du curseur.

En effet, comme le programme peut fonctionner 'à vide' et afin de bien marquer l'ouverture – ou la non-ouverture d'un fichier, les curseurs ont des formes différentes selon les cas :

- curseur en forme de roue si un fichier est ouvert (identique à celui employé pour l'accessoire de cryptage "Kruptos" du précédent numéro de Pom's) ;
- curseur en forme de 'sens interdit' si il n'y a pas de fichier ouvert.

Un éventuel fichier ouvert sera aussi fermé si :

- le choix d'un autre fichier est effectué ;
- l'utilisateur quitte le programme.

Numéros

Si l'on veut se constituer un fichier d'adresses, on souhaitera conserver les numéros de téléphone. Cependant, les numéros de téléphones sont parfaitement inutiles lorsqu'il s'agit de faire un 'mailing'. TPom's permet donc, grâce à l'article 'Numéros', de choisir le format de stockage : avec ou sans numéros.

Dans le cas d'un stockage sans numéro, le format du fichier comprend quatre zones séparées par un 'retour-chariot' :

```
1 Nom
2 Prénom
3 Adresse
4 Ville
```

Pour un stockage avec numéros, le format comprend cinq zones :

```
1 Nom
2 Prénom
3 Adresse
4 Ville
5 Numéro de téléphone
```

Afin de faciliter le traitement du fichier d'adresses par un éventuel autre programme, chaque 'fiche' est précédée par une 'puce' (•), et ce quel que soit le format utilisé (avec ou sans numéros), ce qui nous donne par exemple, sans numéros :

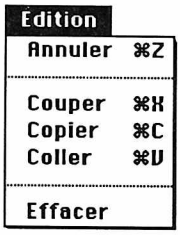
```
•
Electricité de France-Gaz de
  France(E.D.F-G.D.F)

46 bd St Antoine
LE CHESNAY
•
Electricité de France Gaz de
```

```
France

7 bd Jean Jaurès
HOUILLES
•
Société Agricole d'Electricité
  de la Région d'Orgerus

33 r Gare
TACOIGNIERES
```

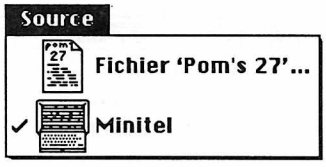
et, avec numéros :

```
•
Electricité de France-Gaz de
  France(E.D.F-G.D.F)

46 bd St Antoine
LE CHESNAY
(1) 39 54 90 10
•
Electricité de France Gaz de
  France

7 bd Jean Jaurès
HOUILLES
(1) 39 13 24 45
•
Société Agricole d'Electricité
  de la Région d'Orgerus

33 r Gare
TACOIGNIERES
(1) 34 87 22 17
```

Lorsque l'option 'avec numéros' est valide, l'article correspondant apparaît coché et, pendant le traitement, les numéros de téléphone s'affichent à l'écran. Par défaut, les numéros ne sont pas conservés. Notons enfin que, comme il est possible de changer de format à tout instant, un même fichier peut contenir des 'fiches' avec et sans numéros de téléphone.



TPom's n'a pas recours au 'Couper/Copier/Coller' ; le menu édition n'est donc là que pour répondre aux exigences – légitimes – du 'User Interface Guideline', et ne deviendra valide qu'en présence d'un accessoire de bureau actif.

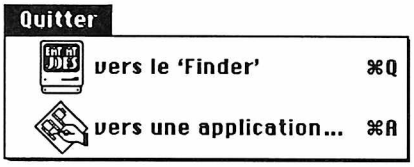


Fichier 'Pom's 27'...

... sera utilisé si l'on désire traiter un fichier créé par le programme du numéro 27 de Pom's. La fenêtre standard de sélection s'affiche, ne laissant apparaître que les noms des fichiers cités plus haut. Une fois la sélection faite, TPom's fait rapidement son travail, ne 'rendant la main' qu'à l'achèvement de la tâche.

Minitel

Dans ce cas, les informations traitées proviennent du Minitel, via l'interface série 'communication' du Macintosh. Ce sera certainement la manière la plus courante d'employer TPom's ; c'est pourquoi elle est celle valide par défaut.



TPom's offre deux façon de sortir du programme :

- en provoquant un retour au 'Finder' (méthode courante) ;
- en proposant l'appel direct de l'application de votre choix. Il n'est pas nécessaire que l'application à appeler soit sur un volume 'en ligne'.

Il est possible d'utiliser les équivalents claviers Commande-Q et Commande-A.

Problèmes possibles

Prise en compte des informations

TPom's ne prend en compte les informations issues du Minitel – ou puisées dans un fichier 'Pom's 27' – que lorsqu'il reçoit le premier nom de ville. Autrement dit, si vous lancez TPom's pendant que vous consultez, par exemple, la liste des plombiers de Paris, il n'y a aucune chance pour que le nom de la ville change, et notre programme ne va rien retenir. Pour éviter ce genre de situation, il est préférable de faire démarrer TPom's avant de commencer la consultation de l'annuaire électronique. Une autre méthode permet de se sortir d'embarras : appuyer sur la touche répétition, ce qui oblige le serveur des PTT à réémettre, entre autre, le nom de la ville, déclenchant ainsi la prise en compte des informations.

Informations erronées

L'annuaire électronique ne propose pas de zones fixes – sauf pour le numéro de téléphone – dans lesquelles TPom's aurait pu puiser. Notre programme fait donc de son mieux en analysant la multitude de codes qui lui parviennent. Dans ces conditions, il peut arriver (en fait, il arrive !) que certaines informations stockées dans le fichier texte soit erronées. De ce fait, un petit coup d'œil dans le fichier avec un logiciel de traitement de texte peut s'avérer utile.

Zones

Bien qu'étant affichés dans la fenêtre pendant le traitement, les départements ne sont pas enregistrés, ceux-ci ne présentant pas d'intérêt.

L'annuaire électronique n'indiquant pas le code postal, il sera nécessaire, pour un 'mailing' par exemple, de faire un remplacement systématique VILLE — CODE POSTAL / VILLE.

Mauvaise connexion

Le programme aura des difficultés à fonctionner si le câble de liaison est mal – ou pas – connecté. Afin d'éviter cela, le programme utilise des curseurs

qui doivent s'animer à la réception du moindre caractère. Si tel n'est pas le cas, une vérification de la 'connectique' s'impose.

Fichier PTT

Nom : PTT Paris
Prénom : comédiens des PTT
Adresse : 181 av Choisy 13e
Ville : PARIS
Département : PARIS
Numéro : (1) 47 07 57 42

Sources et fichiers du programme "T_Pom's"

* le symbole '∫' indique la continuité d'une ligne.

Fichier 'T_Pom's.Job'

```
ASM      T_Pom's.Asm   Exec   Edit
LINK     T_Pom's.Link  Exec   Edit
RMAKER   T_Pom's.R     Finder Edit
```

Fichier 'T_Pom's.Link'

```
]

T_Pom's
/Output T_Pom's.Code
/Type 'TEMP'

$
```

Source 'T_Pom's.Asm'

```
    INCLUDE  T_Pom's/1.Asm
    INCLUDE  T_Pom's/2.Asm
    INCLUDE  T_Pom's/3.Asm
    INCLUDE  T_Pom's/4.Asm
    END
```

Source 'T_Pom's/1.Asm'

```
INCLUDE      FSEqu.Txt
INCLUDE      Traps.Txt
INCLUDE      SysEqu.D
INCLUDE      ToolEqu.D
INCLUDE      QuickEqu.D
INCLUDE      PackMacs.Txt

MenuPomme    EQU   1
 mAPropos    EQU   1
MenuFichier  EQU   2
 mNouveau    EQU   1
 mFermer     EQU   3
 mNumero     EQU   5
MenuEdition  EQU   3
 mAnnuler    EQU   1
 mCouper     EQU   3
 mCopier     EQU   4
 mColler     EQU   5
 mEffacer    EQU   7
MenuSource   EQU   4
 mPoms27     EQU   1
 mMinitel    EQU   2

MenuSortie   EQU   5
 mQuitter    EQU   1
 mApplication EQU  2

DialogAPropos EQU  1
DialogBis    EQU   501
bOK          EQU   1

Zero         EQU   0
Un           EQU   1
Deux         EQU   2
Trois        EQU   3
Quatre       EQU   4

Debut
    BSR.S    Initialisations
    BSR      Menus
    BSR      OuvreSerie
    BSR      OuvertureF
    BRA      Evenements

Initialisations
    PEA      -4(A5)
    _InitGraf
    _InitFonts
    MOVE.L   #$0000FFFF,D0
    _FlushEvents
    _InitWindows
    _InitMenus
    CLR.L    -(SP)
    _InitDialogs
    _TEInit
    PEA      Curseur1bis
    _SetCursor

    SUBQ.L   #4,SP
    _TickCount
    MOVE.L   (SP)+,∫
             CompteurTiming(A5)
    MOVE     #1,Numero_Curseur(A5)
    SF       DrMenuFichier(A5)
    ST       DrMinitel(A5)
    ST       DrCom1(A5)
    ST       DrLPT1Valide(A5)
    SF       DrNumero(A5)
    SF       DrDepart(A5)
    SF       DrCouleur(A5)
    SF       DrNumero_en_cours(A5)
    SF       DrLigne(A5)
    SF       DrColonne(A5)
    SF       DrRepere(A5)
    SF       DrAttente_Affichage_∫
             Ville(A5)
    SF       Dr_Attente_Ville(A5)
    SF       Dr_Attente_∫
             Departement(A5)
    SF       Dr_Attente_Prenom(A5)
    SF       Dr_Attente_∫
             Nom_Suite(A5)
    SF       Dr_Attente_Adresse(A5)
    SF       DrAffichage(A5)
    SF       Copie_Affichage(A5)
    SF       Copie_Affichage+128(A5)
    SF       Copie_Affichage+256(A5)
    SF       Copie_Affichage+384(A5)
    SF       Copie_Affichage+448(A5)
    SF       Copie_Affichage+512(A5)
    BSR      Zero_Chaines_Nom
    MOVE     #DILoad,-(SP)
    _Pack2
    RTS

Menus
    SUBQ.L   #Quatre,SP
    MOVE     #MenuPomme,-(SP)
    _GetRMenu
    MOVE.L   (SP),HandleMPomme(A5)
    MOVE.L   HandleMPomme(A5),-(SP)
    CLR      -(SP)
    _InsertMenu
    MOVE.L   #'DRVR',-(SP)
    _AddResMenu
    SUBQ.L   #Quatre,SP
    MOVE     #MenuFichier,-(SP)
    _GetRMenu
    MOVE.L   (SP),HandleMFichier(A5)
    CLR      -(SP)
    _InsertMenu
    SUBQ.L   #Quatre,SP
    MOVE     #MenuEdition,-(SP)
    _GetRMenu
    MOVE.L   (SP),HandleMEdition(A5)
    CLR      -(SP)
    _InsertMenu
    SUBQ.L   #Quatre,SP
    MOVE     #MenuSource,-(SP)
    _GetRMenu
    MOVE.L   (SP),HandleMSource(A5)
    CLR      -(SP)
    _InsertMenu
    SUBQ.L   #Quatre,SP
    MOVE     #MenuSortie,-(SP)
    _GetRMenu
    MOVE.L   (SP),HandleMSortie(A5)
    CLR      -(SP)
    _InsertMenu
@1  MOVE.L   HandleMEdition(A5),∫
             -(SP)
    CLR      -(SP)
    _DisableItem
    _DrawMenuBar
    RTS

Evenements
    _SystemTask
    BSR      TesteEntree
    BEQ.S    @1
    BSR      Niveau0
    BRA.S    Evenements
@1  SUBQ.L   #2,SP
    MOVE     #$0FFF,-(SP)
    PEA      EnregEvents(A5)
    _GetNextEvent
    TST      (SP)+
    BEQ.S    Evenements
    BSR.S    Traitementevents
    BEQ.S    Evenements
    RTS

Traitementevents
    MOVE     EnregEvents+∫
             evtNum(A5),D0
    ADD      D0,D0
    MOVE     TableEvenements(D0),D0
    JMP      TableEvenements(D0)
```

```
TableEvenements
    DC      Eventsuivant-∫
            TableEvenements
    DC      Souris-TableEvenements
    DC      Eventsuivant-∫
            TableEvenements
    DC      Touche-TableEvenements
    DC      Eventsuivant-∫
            TableEvenements
    DC      Touche-TableEvenements
    DC      MiseJour-TableEvenements
    DC      Disquette-∫
            TableEvenements
    DC      Active-TableEvenements
    DC      Eventsuivant-∫
            TableEvenements
    DC      Eventsuivant-∫
            TableEvenements
    DC      Eventsuivant-∫
            TableEvenements
Eventsuivant
    MOVEQ #Zero,D0
    RTS

Souris
    SUBQ.L  #Deux,SP
    MOVE.L  EnregEvents+∫
            evtMouse(A5),-(SP)
    PEA     EnregEvents+evtMeta+∫
            Deux(A5)
    _FindWindow
    MOVE    (SP)+,D0
    ADD     D0,D0
    MOVE    TableFenetre(D0),D0
    JMP     TableFenetre(D0)
TableFenetre
    DC      Eventsuivant-TableFenetre
    DC      BarMenu-TableFenetre
    DC      fenetreSys-TableFenetre
    DC      Contenu-TableFenetre
    DC      DeplaceFen-TableFenetre
    DC      Contenu-TableFenetre
    DC      Eventsuivant-TableFenetre

BarMenu
    SUBQ.L  #Quatre,SP
    MOVE.L  EnregEvents+∫
            evtMouse(A5),-(SP)
    _MenuSelect
    MOVE    (SP)+,D1
    MOVE    (SP)+,D0
BarMenuC
    ADD     D1,D1
    MOVE    TableMenus(D1),D1
    JMP     TableMenus(D1)
TableMenus
    DC      Eventsuivant-TableMenus
    DC      TraitPomme-TableMenus
    DC      TraitFichier-TableMenus
    DC      TraitEdition-TableMenus
    DC      TraitSource-TableMenus
    DC      TraitSortie-TableMenus

FinMenu
    CLR     -(SP)
    _HiliteMenu
    BRA.S   Eventsuivant

TraitFichier
    ADD     D0,D0
    MOVE    TableFichier(D0),D0
    JMP     TableFichier(D0)
TableFichier
    DC      FinMenu-TableFichier
    DC      TmNouveau-TableFichier
    DC      FinMenu-TableFichier
    DC      TmFermer-TableFichier
    DC      FinMenu-TableFichier
    DC      TmNumero-TableFichier

Touche
    BTST    #CmdKey,EnregEvents+∫
            evtMeta(A5)
    BNE.S   Commande
    BRA     Eventsuivant

Commande
    CLR.L   -(SP)
    MOVE    EnregEvents+evtMessage+∫
            Deux(A5),-(SP)
    _MenuKey
    MOVE    (SP)+,D1
    MOVE    (SP)+,D0
    BRA     BarMenuC

MiseJour
    BSR     SPMiseJour
    BRA     Eventsuivant

SPMiseJour
    MOVEA.L EnregEvents+∫
            evtMessage(A5),A4
    MOVE.L  A4,-(SP)
    MOVE.L  (SP),-(SP)
    _BeginUpdate
    _SetPort
    BSR     AfficheTitre
    BSR     MiseJourZones
    MOVE.L  A4,-(SP)
    _EndUpdate
    RTS

Disquette
    BRA     Eventsuivant

Active
    MOVEA.L EnregEvents+∫
            evtMessage(A5),A4
    CMPI    #userKind,∫
            windowKind(A4)
    BNE     Eventsuivant
    MOVE    EnregEvents+∫
            evtMeta(A5),D7
    BTST    #ActiveFlag,D7
    BEQ.S   DeActiver
    MOVE.L  A4,-(SP)
    _SetPort
    BSR     ChangeCurseur
DeActiver
    BSR     ValideMenus
    BRA     Eventsuivant

fenetreSys
    PEA     EnregEvents(A5)
    MOVE.L  EnregEvents+evtMeta+∫
            Deux(A5),-(SP)
    _SystemClick
    BRA     Eventsuivant

DeplaceFen
    MOVE    EnregEvents+∫
            evtMeta(A5),D0
    ANDI    #2048+256,D0
    CMPI    #2048+256,D0
    BNE.S   @1
    SUBQ.L  #Quatre,SP
    _FrontWindow
    MOVE.L  (SP)+,D3
    MOVE.L  PointeurFenetre(A5),-(SP)
    LEA     RectFenetre,A3
    MOVE.L  (A3),-(SP)
    CMP.L   PointeurFenetre(A5),D3
    SEQ     -(SP)
    _MoveWindow
    BRA     Eventsuivant
@1  MOVE.L  EnregEvents+evtMeta+∫
            Deux(A5),A4
    MOVE.L  A4,-(SP)
    MOVE.L  EnregEvents+∫
            evtMouse(A5),-(SP)
    PEA     Rectanglelimite
    _DragWindow
@2  BRA     Eventsuivant

Contenu
    SUBQ.L  #Quatre,SP
    _FrontWindow
    MOVE.L  (SP)+,D0
    MOVEA.L EnregEvents+evtMeta+∫
            Deux(A5),A4
    CMP.L   A4,D0
    BEQ.S   @1
    MOVE.L  A4,-(SP)
    _SelectWindow
    BRA.S   @2
@1  MOVE.L  A4,-(SP)
    _SetPort
    PEA     EnregEvents+∫
            EvtMouse(A5)
    _GlobalToLocal
    SUBQ.L  #Deux,SP
    MOVE.L  EnregEvents+∫
            EvtMouse(A5),-(SP)
    MOVE.L  A4,-(SP)
    PEA     HandleControle(A5)
    _FindControl
    TST     (SP)+
    BEQ.S   @2
    SUBQ.L  #Deux,SP
    MOVE.L  HandleControle(A5),-(SP)
    MOVE.L  EnregEvents+∫
            EvtMouse(A5),-(SP)
    CLR.L   -(SP)
    _TrackControl
    TST     (SP)+
    BEQ.S   @2
    SUBQ.L  #Quatre,SP
    MOVE.L  HandleControle(A5),-(SP)
    _GetCRefCon
    BSR     Boutons
@2  BRA     Eventsuivant

TraitPomme
    CMPI    #mAPropos,D0
    BEQ.S   TraiteAPropos
    MOVE.L  HandleMPomme(A5),-(SP)
    MOVE    D0,-(SP)
    PEA     NomAccessoire(A5)
    _GetItem
    SUBQ.L  #Deux,SP
    PEA     NomAccessoire(A5)
    _OpenDeskAcc
    ADDQ    #Deux,SP
    BRA     FinMenu

TraiteAPropos
    CLR.L   -(SP)
    MOVE    #DialogAPropos,-(SP)
    PEA     Dialogue(A5)
    MOVE.L  #-1,-(SP)
    _GetNewDialog
    MOVE.L  (SP),-(SP)
    _SetPort
Attente
    CLR.L   -(SP)
    PEA     Article(A5)
    _ModalDialog
    _CloseDialog
    SUBQ.L  #Quatre,SP
    _FrontWindow
    MOVE.L  (SP)+,A4
    CMPI    #userKind,∫
            windowKind(A4)
    BNE.S   @1
    MOVE.L  A4,-(SP)
    MOVE.L  (SP),-(SP)
    _SelectWindow
```

```
    _SetPort
@1  BRA     FinMenu

TraitEdition
    BSR.S   VersSysteme
    BRA     FinMenu

VersSysteme
    MOVE    D0,D7
    CLR     -(SP)
    MOVE    D0,-(SP)
    SUBQ    #Un,(SP)
    _SysEdit
    MOVE.B  (SP)+,D1
    RTS

TmNouveau
    MOVE.L  #$640064,-(SP)
    PEA     Prompt
    PEA     NomDefaut
    CLR.L   -(SP)
    PEA     ReponseGetFile∫
            Destination(A5)
    MOVE    #SFPutFile,-(SP)
    _Pack3
    TST     ReponseGetFile∫
            Destination+rGood(A5)
    BEQ     FinMenu
    TST.B   DrMenuFichier(A5)
    BEQ.S   @1
    LEA     TamponIO∫
            Destination(A5),A0
    _Close
    BSR     InfoFichier
@1  LEA     TamponIO∫
            Destination(A5),A0
    LEA     ReponseGetFileDest∫
            ination+rName(A5),A1
    MOVE.L  A1,ioFileName(A0)
    MOVE    ReponseGetFileDest∫
            ination+rVolume(A5),∫
            ioVRefNum(A0)
    _Create
    BEQ.S   @2
    CMPI    #-48,D0
    BEQ.S   @2
@3  CMPI    #-33,D0
    BNE.S   @4
    MOVE    #2,-(SP)
    BSR     Erreur
    BRA.S   @5
@4  CMPI    #-34,D0
    BNE.S   @6
    MOVE    #3,-(SP)
    BSR     Erreur
    BRA.S   @5
@6  MOVE    #1,-(SP)
    BSR     Erreur
@5  MOVE.L  PointeurFenetre(A5),-(SP)
    PEA     Titre
    _SetWTitle
    MOVE.L  HandleMFichier(A5),∫
            -(SP)
    MOVE    #mFermer,-(SP)
    _DisableItem
    SF      DrMenuFichier(A5)
    BSR     ChangeCurseur
    BRA     FinMenu
@2  LEA     TamponIO∫
            Destination(A5),A0
    LEA     ReponseGetFileDest∫
            ination+rName(A5),A1
    MOVE.L  A1,ioFileName(A0)
    CLR.L   ioOwnBuf(A0)
    MOVE.B  #fsWrPerm,ioPermssn(A0)
    MOVE    ReponseGetFile∫
            Destination+rVolume∫
            (A5),ioVRefNum(A0)
    _Open
    BNE.S   @3
    MOVE.L  PointeurFenetre(A5),-(SP)
    PEA     ReponseGetFileDest∫
            ination+rName(A5)
    _SetWTitle
    MOVEQ   #0,D0
    MOVE.B  ReponseGetFileDest∫
            ination+rName(A5),D0
    ADDQ    #1,D0
    LEA     ReponseGetFileDest∫
            ination+rName(A5),A0
    LEA     NomFichier(A5),A1
    _BlockMove
    MOVE    ReponseGetFile∫
            Destination+rVolume∫
            (A5),Volume(A5)
    ST      DrMenuFichier(A5)
    BSR     ChangeCurseur
    MOVE.L  HandleMFichier(A5),∫
            -(SP)
    MOVE    #mFermer,-(SP)
    _EnableItem
    BRA     FinMenu

TmFermer
    SUBQ.L  #Quatre,SP
    _FrontWindow
    MOVEA.L (SP)+,A4
    CMPA.L  PointeurFenetre(A5),A4
    BEQ.S   @1
    MOVE    windowKind(A4),-(SP)
    _CloseDeskAcc
    BRA     FinMenu
@1  BSR     FermeDestination
    BRA     FinMenu

OuvertureF
    SUBQ.L  #Quatre,SP
    CLR.L   -(SP)
    PEA     RectFenetre
    PEA     Titre
    MOVE.B  #Un,-(SP)
    MOVE    #documentProc,-(SP)
    CLR.L   -(SP)
    CLR     -(SP)
    MOVEQ   #Un,D0
    MOVE.L  D0,-(SP)
    _NewWindow
    MOVEA.L (SP),A4
    _SetPort
    MOVE    #sysFont,-(SP)
    _TextFont
    MOVE    #12,-(SP)
    _TextSize
    MOVE.L  A4,PointeurFenetre(A5)
    RTS

OuvreSerie
    PEA     TamponIOSort(A5)
    PEA     Sort
    BSR     OuvreS
    PEA     TamponIOEntre(A5)
    PEA     Entre
    BSR     OuvreS
    RTS

OuvreS
    LINK    A6,#0
    MOVEA.L 12(A6),A4
    MOVEA.L A4,A0
    MOVE.L  8(A6),ioFileName(A0)
    CLR.B   ioPermssn(A0)
    _Open
    BNE.S   @1
    MOVEA.L A4,A0
    MOVE    #8,csCode(A0)
    MOVE    #94+1024+16384+12288,∫
            csParam(A0)
    _Control,Immed
    BNE.S   @1
    MOVE.L  #800,D0
    _NewPtr,clear
    BNE.S   @1
    MOVE.L  A0,A1
    MOVEA.L A4,A0
    MOVE    #9,csCode(A0)
    MOVE.L  A1,csParam(A0)
    MOVE    #800,csParam+4(A0)
    _Control,Immed
    BEQ.S   @2
@1  MOVE    #2,-(SP)
    BSR     Erreur
    _ExitToShell
@2  UNLK    A6
    MOVE.L  (SP),8(SP)
    ADDQ.L  #8,SP
    RTS

AfficheChaineNumero
    MOVE.L  #$0060002E,-(SP)
    _MoveTo
    PEA     ChaineNumero
    _DrawString
    RTS

AfficheTitre
    MOVE.L  #$00100044,-(SP)
    _MoveTo
    PEA     ChaineNom
    _DrawString
    MOVE.L  #$0020002F,-(SP)
    _MoveTo
    PEA     ChainePrenom
    _DrawString
    MOVE.L  #$0030002D,-(SP)
    _MoveTo
    PEA     ChaineAdresse
    _DrawString
    MOVE.L  #$00400045,-(SP)
    _MoveTo
    PEA     ChaineVille
    _DrawString
    MOVE.L  #$0050000B,-(SP)
    _MoveTo
    PEA     ChaineDepartement
    _DrawString
    TST.B   DrNumero(A5)
    BEQ.S   @1
    BSR     AfficheChaineNumero
@1  RTS

RectFenetre     DC      50,8,154,478
Titre           DC.B    21,'Pas de ∫
                        fichier ouvert',0
                .Align  2
TypeTEXT        DC.B    'TEXT'
TypeAPPL        DC.B    'APPL'
CreateurWrite   DC.B    'MACA'
Type27          DC.B    'mtlF'
Dialogue        DS.B    DWindLen
NomAccessoire   DS.B    32
EnregEvents     DS.B    20
HandleMPomme    DS.L    1
HandleMFichier  DS.L    1
HandleMEdition  DS.L    1
HandleMSource   DS.L    1
HandleMSortie   DS.L    1
Article         DS      1
SauvePort       DS.L    1
PointeurFenetre DS.L    1
NomFichier      DS.B    64
Volume          DS      1
TamponIO        DS.B    ioFQElSize
TamponIOFichier27
                DS.B    ioFQElSize
TamponIODestination
                DS.B    ioFQElSize
ReponseGetFile  DS.B    72
```

```
ReponseGetFileDestination
                DS.B    72
ReponseGetFileQuitter
                DS.B    72
Entre           DC.B    4,'.AIn',0
Sort            DC.B    5,'.AOut'
TamponIOEntre   DS.B    ioFQElSize
TamponIOSort    DS.B    ioFQElSize
TamponSortie    DS.B    700
TamponEntre     DS.B    700
BufferErreur    DS      5
VolumeCourant   DS      1
Rectanglelimite DC      28,4,338,508
RectEffaceNumero
                DC      84,0,100,500
RectEffaceVille DC      0,104,100,500
RectEffaceNom1  DC      0,104,52,500
RectEffaceNom2  DC      84,104,100,500
HandleControle  DS.L    1
PointeurTampon  DS.L    1
CompteurTiming  DS.L    1
Numero_Curseur  DS      1
C_Nom           DS.B    128
C_Prenom        DS.B    128
C_Adresse       DS.B    128
C_Ville         DS.B    64
C_Departement   DS.B    64
C_Numero        DS.B    32
Copie_Affichage DS.B    544
DrMenuFichier   DS.B    1
DrMenuEdition   DS.B    1
DrMenuMode      DS.B    1
DrLPT1Valide    DS.B    1
DrMinitel       DS.B    1
DrCom1          DS.B    1
DrNumero        DS.B    1
DrDepart        DS.B    1
DrCouleur       DS.B    1
DrAttente_Affichage_Ville
                DS.B    1
DrRepere        DS.B    1
Dr_Attente_Ville
                DS.B    1
DrNumero_en_cours
                DS.B    1
DrLigne         DS.B    1
DrColonne       DS.B    1
Dr_Attente_Nom  DS.B    1
Dr_Attente_Departement
                DS.B    1
Dr_Attente_Prenom
                DS.B    1
Dr_Attente_Nom_Suite
                DS.B    1
Dr_Attente_Adresse
                DS.B    1
DrAffichage     DS.B    1
DrFinFichier    DS.B    1
Dr_Erreur       DS.B    1
ChaineNom       DC.B    5,'Nom :'
ChainePrenom    DC.B    8,'Prénom :',0
ChaineAdresse   DC.B    9,'Adresse :'
ChaineVille     DC.B    7,'Ville :'
ChaineDepartement
                DC.B    13,∫
                        'Département :'
ChaineNumero    DC.B    8,'Numéro :',0
Prompt          DC.B    18,'Enregistrer ∫
                        sous :',0
NomDefaut       DC.B    10,'Sans titre',0
Table1          DC.B    $23,'£'
                DC.B    $24,'$'
                DC.B    $26,'#'
                DC.B    $27,'§'
                DC.B    $2C,'<'
                DC.B    $2D,'^'
                DC.B    $2E,'>'
                DC.B    $2F,'v'
                DC.B    $30,'°'
                DC.B    $31,'±'
                DC.B    $38,'÷'
                DC.B    $3C,'/'
                DC.B    $3D,'/'
                DC.B    $3E,'/'
                DC.B    $6A,'Œ'
                DC.B    $7A,'œ'
                DC.B    $8A,'ß'
Table_Gra       DC.B    'aàeèuù'
Table_Cir       DC.B    'aâeêiîoôuû'
Table_Tre       DC.B    'aäeëiïoöuü'
                .Align  2
Curseur1        DC.L    $07C01930,$21084104
                DC.L    $41048102,$8102FFFE
                DC.L    $81028102,$41044104
                DC.L    $21081930,$07C00000
                DC.L    0,0,0,0,0,0,0,0
                DC      7,7
Curseur2        DC.L    $07C01830,$20484044
                DC.L    $4084B082,$8D028382
                DC.L    $8162821A,$42044404
                DC.L    $24081830,$07C00000
                DC.L    0,0,0,0,0,0,0,0
                DC      7,7
Curseur3        DC.L    $07C01830,$20085014
                DC.L    $48248442,$82828102
                DC.L    $82828442,$48245014
                DC.L    $20081830,$07C00000
                DC.L    0,0,0,0,0,0,0,0
                DC      7,7
Curseur4        DC.L    $07C01830,$24084404
                DC.L    $4204821A,$81628382
                DC.L    $8D02B082,$40844044
                DC.L    $20481830,$07C00000
                DC.L    0,0,0,0,0,0,0,0
                DC      7,7
Curseur1bis     DC.L    $07C01FF0,$3FF87FFC
                DC.L    $7FFCFFFE,$C006C006
                DC.L    $C006FFFE,$7FFC7FFC
                DC.L    $3FF81FF0,$07C00000
                DC.L    $07C01FF0,$3FF87FFC
                DC.L    $7FFCFFFE,$FFFEFFFE
                DC.L    $FFFEFFFE,$7FFC7FFC
                DC.L    $3FF81FF0,$07C00000
                DC      7,7
Curseur2bis     DC.L    $07C01FF0,$3FF87FFC
                DC.L    $7FFCFFFE,$FFFE8002
                DC.L    $FFFEFFFE,$7FFC7FFC
                DC.L    $3FF81FF0,$07C00000
                DC.L    $07C01FF0,$3FF87FFC
                DC.L    $7FFCFFFE,$FFFEFFFE
                DC.L    $FFFEFFFE,$7FFC7FFC
                DC.L    $3FF81FF0,$07C00000
                DC      7,7
```

Source 'T_Pom's/2.Asm'

```
InfoFichier
    LEA         TamponIO∫
                Destination(A5),A0
    LEA         NomFichier(A5),A1
    MOVE.L      A1,ioFileName(A0)
    MOVE        Volume(A5),∫
                ioVRefNum(A0)
    CLR.B       ioFileType(A0)
    CLR         ioFDirIndex(A0)
    _GetFileInfo
    BEQ.S       @1
@2  MOVE        #1,-(SP)
    BSR         Erreur
    RTS

@1  LEA         TamponIO∫
                Destination(A5),A0
    MOVE.L      #'TEXT',∫
                ioFlUsrWds+fdType(A0)
    MOVE.L      #'MACA',ioFlUsrWds+∫
                fdCreator(A0)
    _SetFileInfo
    BNE.S       @2
    RTS

FermeDestination
    SF          DrMenuFichier(A5)
    BSR         ChangeCurseur
    LEA         TamponIO∫
                Destination(A5),A0
    _Close
    BSR         InfoFichier
    MOVE.L      PointeurFenetre(A5),-(SP)
    PEA         Titre
    _SetWTitle
    MOVE.L      HandleMFichier(A5),∫
                -(SP)
    MOVE        #mFermer,-(SP)
    _DisableItem
    RTS

Boutons
    RTS

Beep7
    MOVE        #7,-(SP)
    _SysBeep
    RTS

Erreur
    LINK        A6,#0
    BSR         Beep7
    CLR.L       -(SP)
    MOVE.L      #'STR ',-(SP)
    MOVE        8(A6),-(SP)
    _GetResource
    MOVE.L      (SP)+,A2
    MOVE.L      A2,A0
    _HLock
    MOVE.L      (A2),-(SP)
    CLR.L       -(SP)
    CLR.L       -(SP)
    CLR.L       -(SP)
    _ParamText
    CLR         -(SP)
    MOVE        #128,-(SP)
    CLR.L       -(SP)
    _StopAlert
    MOVE        (SP)+,D0
    MOVE.L      A2,A0
    _HUnLock
    UNLK        A6
    MOVE.L      (SP),Deux(SP)
    ADDQ.L      #Deux,SP
    RTS

TesteEntree
    LEA         TamponIOEntre(A5),A0
    MOVE        #2,csCode(A0)
    _Status,Immed
    TST.L       csParam(A0)
    RTS

ValideMenus
    SUBQ.L      #4,SP
    _FrontWindow
    MOVEA.L     (SP)+,A4
    MOVEQ       #0,D5
    CMP         #UserKind,∫
                windowKind(A4)
    BNE.S       FenetreAccessoire
    TST.B       DrMenuFichier(A5)
    BNE.S       @1
    MOVE.L      HandleMFichier(A5),∫
                -(SP)
    MOVE        #mFermer,-(SP)
    _DisableItem
@1  MOVE.L      HandleMFichier(A5),∫
                -(SP)
    MOVE        #mNouveau,-(SP)
```

```
    _EnableItem
    MOVE.L    HandleMEdition(A5),∫
              -(SP)
    CLR       -(SP)
    _DisableItem
    _DrawMenuBar
    RTS

FenetreAccessoire
    TST.B     DrMenuFichier(A5)
    BNE.S     @1
    MOVE.L    HandleMFichier(A5),∫
              -(SP)
    MOVE      #mFermer,-(SP)
    _EnableItem
@1  MOVE.L    HandleMEdition(A5),∫
              -(SP)
    CLR       -(SP)
    _EnableItem
    _DrawMenuBar
    RTS

LireCaractere
    TST.B     DrMinitel(A5)
    BNE.S     @1
    LEA       TamponIO∫
              Fichier27(A5),A0
    MOVE.L    #Un,ioReqCount(A0)
    LEA       scratch8,A3
    MOVE.L    A3,ioBuffer(A0)
    MOVE      #fsAtMark,ioPosMode(A0)
    CLR.L     ioPosOffset(A0)
    _Read
    BEQ.S     @2
    CMPI      #-39,D0
    BEQ.S     @3
    MOVE      #1,-(SP)
    BSR       Erreur
@3  ST        DrFinFichier(A5)
    RTS
@1  LEA       TamponIOEntre(A5),A0
    MOVE.L    #Un,ioReqCount(A0)
    LEA       scratch8,A3
    MOVE.L    A3,ioBuffer(A0)
    MOVE      #fsFromStart,∫
              ioPosMode(A0)
    CLR.L     ioPosOffset(A0)
    _Read
@2  MOVE.B    (A3),D5
    ANDI      #$007F,D5
    RTS

Place_Caractere
    LINK      A6,#0
    MOVE.L    8(A6),A0
    ADDQ.B    #1,(A0)
    MOVE.B    (A0),D0
    EXT       D0
    MOVE.B    D5,0(A0,D0)
    UNLK      A6
    MOVE.L    (SP),4(SP)
    ADDQ.L    #4,SP
    RTS

Enregistre
    TST.B     DrMenuFichier(A5)
    BEQ       @2
    LEA       C_Nom(A5),A0
    LEA       Copie_Affichage(A5),A1
    MOVE.L    #544,D0
    _BlockMove
    MOVE.B    #'•',Scratch8
    PEA       Scratch8
    MOVEQ     #1,D0
    MOVE.L    D0,-(SP)
    BSR       Ecrit
    TST.B     Dr_Erreur(A5)
    BNE       @2
    BSR       Ecrit_CR
    TST.B     Dr_Erreur(A5)
    BNE       @2
    MOVE.B    C_Nom(A5),D0
    EXT       D0
    EXT.L     D0
    PEA       C_Nom+1(A5)
    MOVE.L    D0,-(SP)
    BSR       Ecrit
    TST.B     Dr_Erreur(A5)
    BNE       @2
    BSR       Ecrit_CR
    TST.B     Dr_Erreur(A5)
    BNE       @2
    TST.B     C_Prenom(A5)
    BEQ.S     @1
    MOVE.B    C_Prenom(A5),D0
    EXT       D0
    EXT.L     D0
    PEA       C_Prenom+1(A5)
    MOVE.L    D0,-(SP)
    BSR       Ecrit
    TST.B     Dr_Erreur(A5)
    BNE.S     @2
@1  BSR       Ecrit_CR
    TST.B     Dr_Erreur(A5)
    BNE.S     @2
    MOVE.B    C_Adresse(A5),D0
    EXT       D0
    EXT.L     D0
    PEA       C_Adresse+1(A5)
    MOVE.L    D0,-(SP)
    BSR       Ecrit
    TST.B     Dr_Erreur(A5)
    BNE.S     @2
    BSR       Ecrit_CR
    TST.B     Dr_Erreur(A5)
    BNE.S     @2
    MOVE.B    C_Ville(A5),D0
    EXT       D0
    EXT.L     D0
    PEA       C_Ville+1(A5)
    MOVE.L    D0,-(SP)
    BSR       Ecrit
    TST.B     Dr_Erreur(A5)
    BNE.S     @2
    BSR       Ecrit_CR
    TST.B     Dr_Erreur(A5)
    BNE.S     @2
    TST.B     DrNumero(A5)
    BEQ.S     @2
    MOVE.B    C_Numero(A5),D0
    EXT       D0
    EXT.L     D0
    PEA       C_Numero+1(A5)
    MOVE.L    D0,-(SP)
    BSR       Ecrit
    TST.B     Dr_Erreur(A5)
    BNE.S     @2
    BSR       Ecrit_CR
@2  BSR       Zero_Chaines_Nom
    RTS

Ecrit_CR
    MOVE.B    #13,Scratch8
    PEA       Scratch8
    MOVEQ     #1,D0
    MOVE.L    D0,-(SP)
    BSR       Ecrit
    RTS

Ecrit
    LINK      A6,#0
    LEA       TamponIO∫
              Destination(A5),A0
    MOVE.L    8(A6),ioReqCount(A0)
    MOVE.L    12(A6),ioBuffer(A0)
    MOVE      #fsAtMark,ioPosMode(A0)
    CLR.L     ioPosOffset(A0)
    _Write
    BEQ.S     @1
    CMPI      #-33,D0
    BNE.S     @2
    MOVE      #2,-(SP)
    BSR       Erreur
@5  ST        Dr_Erreur(A5)
    BRA.S     @3
@2  CMPI      #-34,D0
    BNE.S     @4
    MOVE      #3,-(SP)
    BSR       Erreur
    BRA.S     @5
@4  MOVE      #1,-(SP)
    BSR       Erreur
    BRA.S     @5
@1  SF        Dr_Erreur(A5)
@3  UNLK      A6
    MOVE.L    (SP),8(SP)
    ADDQ.L    #8,SP
    RTS

Zero_Chaines_Nom
    SF        C_Nom(A5)
    SF        C_Prenom(A5)
    SF        C_Adresse(A5)
    SF        C_Numero(A5)
    RTS

Zero_Chaines_Ville
    SF        C_Ville(A5)
    SF        C_Departement(A5)
    RTS

Affichage_Ville
    PEA       RectEffaceVille
    _EraseRect
    MOVE.L    #$0040006D,-(SP)
    _MoveTo
    PEA       C_Ville(A5)
    _DrawString
    MOVE.L    #$0050006D,-(SP)
    _MoveTo
    PEA       C_Departement(A5)
    _DrawString
    LEA       C_Ville(A5),A0
    LEA       Copie_Affichage+∫
              384(A5),A1
    MOVE.L    #128,D0
    _BlockMove
    SF        Copie_Affichage(A5)
    SF        Copie_Affichage+128(A5)
    SF        Copie_Affichage+256(A5)
    SF        Copie_Affichage+512(A5)
    RTS

Affiche_Tous
    MOVEQ     #0,D1
    MOVE.B    C_Nom(A5),D1
    EXT       D1
    ADD       D1,D0
    MOVE.B    C_Prenom(A5),D1
    EXT       D1
    ADD       D1,D0
    MOVE.B    C_Adresse(A5),D1
    EXT       D1
    ADD       D1,D0
    MOVE.B    C_Numero(A5),D1
    EXT       D1
    ADD       D1,D0
    BNE.S     @3
    SF        Dr_Erreur(A5)
    BRA.S     @2
@3  TST.B     DrAffichage(A5)
    BEQ.S     @4
    SF        Dr_Erreur(A5)
    BRA.S     @2
@4  ST        DrAffichage(A5)
    PEA       RectEffaceNom1
    _EraseRect
```

```
    MOVE.L    #$0010006D,-(SP)
    _MoveTo
    PEA       C_Nom(A5)
    _DrawString
    MOVE.L    #$0020006D,-(SP)
    _MoveTo
    PEA       C_Prenom(A5)
    _DrawString
    MOVE.L    #$0030006D,-(SP)
    _MoveTo
    PEA       C_Adresse(A5)
    _DrawString
    TST.B     DrNumero(A5)
    BNE.S     @1
    BSR       Enregistre
@2  RTS
@1  PEA       RectEffaceNom2
    _EraseRect
    MOVE.L    #$0060006D,-(SP)
    _MoveTo
    PEA       C_Numero(A5)
    _DrawString
    BSR       Enregistre
    RTS

Ville_Departement
    TST.B     Dr_Attente_Ville(A5)
    BNE.S     @1
@2  RTS
@1  CMPI.B    #$42,DrCouleur(A5)
    BNE.S     @2
    CMPI.B    #1,DrLigne(A5)
    BEQ.S     @2
    SF        Dr_Attente_Ville(A5)
    ST        Dr_Attente_∫
              Departement(A5)
    RTS

Ville_Ville
    TST.B     Dr_Attente_Ville(A5)
    BNE.S     @1
@2  RTS
@1  CMPI.B    #$47,DrCouleur(A5)
    BNE.S     @2
    SF        C_Ville(A5)
    RTS

Departement_Ligne
    TST.B     Dr_Attente_∫
              Departement(A5)
    BNE.S     @1
    RTS
@1  SF        Dr_Attente_∫
              Departement(A5)
    ST        DrAttente_Affichage_∫
              Ville(A5)
    RTS

Nom_Prenom
    TST.B     Dr_Attente_Nom(A5)
    BEQ.S     @1
    CMPI.B    #6,DrColonne(A5)
    BMI.S     @1
    CMPI.B    #31,DrColonne(A5)
    BGE.S     @1
    SF        Dr_Attente_Nom(A5)
    ST        Dr_Attente_Prenom(A5)
    SF        DrAffichage(A5)
    SF        C_Prenom(A5)
@1  RTS

SuiteNom_Adresse
    TST.B     Dr_Attente_Nom_∫
              Suite(A5)
    BNE.S     @1
@2  RTS
@1  CMPI.B    #$42,DrCouleur(A5)
    BNE.S     @2
    SF        Dr_Attente_Nom_∫
              Suite(A5)
    ST        Dr_Attente_Adresse(A5)
    SF        C_Adresse(A5)
    RTS

Adresse_Fin
    TST.B     Dr_Attente_Adresse(A5)
    BNE.S     @1
@2  RTS
@1  CMPI.B    #$42,DrCouleur(A5)
    BNE.S     @2
    SF        Dr_Attente_Adresse(A5)
    RTS

MiseJourZones
    MOVE.L    #$0040006D,-(SP)
    _MoveTo
    PEA       Copie_Affichage+384(A5)
    _DrawString
    MOVE.L    #$0050006D,-(SP)
    _MoveTo
    PEA       Copie_Affichage+448(A5)
    _DrawString
    MOVE.L    #$0010006D,-(SP)
    _MoveTo
    PEA       Copie_Affichage(A5)
    _DrawString
    MOVE.L    #$0020006D,-(SP)
    _MoveTo
    PEA       Copie_Affichage+128(A5)
    _DrawString
    MOVE.L    #$0030006D,-(SP)
    _MoveTo
    PEA       Copie_Affichage+256(A5)
    _DrawString
    TST.B     DrNumero(A5)
    BNE.S     @1
    RTS
@1  MOVE.L    #$0060006D,-(SP)
    _MoveTo
    PEA       Copie_Affichage+512(A5)
    _DrawString
    RTS
```

Source 'T_Pom's/3.Asm'

```
TraitSource
    ADD       D0,D0
    MOVE      TableSource(D0),D0
    JMP       TableSource(D0)
TableSource
    DC        FinMenu-TableSource
    DC        TmPoms27-TableSource
    DC        TmMinitel-TableSource

TraitSortie
    ADD       D0,D0
    MOVE      TableSortie(D0),D0
    JMP       TableSortie(D0)
TableSortie
    DC        FinMenu-TableSortie
    DC        TmQuitter-TableSortie
    DC        TmApplications-∫
              TableSortie

TmMinitel
    TST.B     DrMinitel(A5)
    BNE.S     @1
    ST        DrMinitel(A5)
    MOVE.L    HandleMSource(A5),-(SP)
    MOVE      #mMinitel,-(SP)
    MOVE.B    #255,-(SP)
    _CheckItem
    MOVE.L    HandleMSource(A5),-(SP)
    MOVE      #mPoms27,-(SP)
    CLR       -(SP)
    _CheckItem
@1  BRA       FinMenu

TmPoms27
    MOVE.L    #$640064,-(SP)
    CLR.L     -(SP)
    PEA       FiltreFichiers
    MOVE      #1,-(SP)
    PEA       Type27
    CLR.L     -(SP)
    PEA       ReponseGetFile(A5)
    MOVE      #SFGetFile,-(SP)
    _Pack3
    BSR       AfficheTitre
    BSR       MiseJourZones
    TST       ReponseGetFile+∫
              rGood(A5)
    BEQ.S     FinMenu
    LEA       TamponIOFichier27(A5),∫
              A0
    LEA       ReponseGetFile+∫
              rName(A5),A1
    MOVE.L    A1,ioFileName(A0)
    MOVE      ReponseGetFile+rVolume∫
              (A5),ioVRefNum(A0)
    CLR.L     ioOwnBuf(A0)
    MOVE.B    #fsRdPerm,ioPermssn(A0)
    _Open
    BEQ.S     @1
    MOVE      #1,-(SP)
    BSR       Erreur
    BRA       FinMenu
@1  SF        DrFinFichier(A5)
    SF        DrMinitel(A5)

Boucle27
    BSR       Niveau0
    TST.B     DrFinFichier(A5)
    BEQ.S     Boucle27
    LEA       TamponIOFichier27(A5),∫
              A0
    _Close
    ST        DrMinitel(A5)
    BRA       FinMenu

TmQuitter
    BSR       FermeSerie
    MOVE      #-1,D0
    RTS

TmApplications
    MOVE      #80,-(SP)
    MOVE      (SP),-(SP)
    CLR.L     -(SP)
    CLR.L     -(SP)
    MOVE      #Un,-(SP)
    PEA       TypeAPPL
    CLR.L     -(SP)
    PEA       ReponseGetFile∫
              Quitter(A5)
    MOVE      #SFGetFile,-(SP)
    _Pack3
    TST       ReponseGetFileQuitter+∫
              rGood(A5)
    BEQ       FinMenu
    BSR       FermeSerie
@3  SUBQ.L    #Quatre,SP
    _FrontWindow
    TST.L     (SP)
    BNE.S     @1
    ADDQ      #Quatre,SP
    BRA.S     @2
@1  _HideWindow
    BRA.S     @3
@2  LEA       TamponIO(A5),A0
    MOVE      ReponseGetFileQuitter∫
              +rVolume(A5),∫
              ioVDrvNum(A0)
    _SetVol
    MOVEQ.L   #Quatre,D0
    MOVE.L    AppParmHandle,A0
    _SetHandleSize
```

```
    MOVE.L    AppParmHandle,A0
    MOVE.L    (A0),A0
    CLR.L     (A0)
    CLR       -(SP)
    PEA       ReponseGetFileQuitter+∫
              rName(A5)
    MOVE.L    SP,A0
    _Launch

FermeSerie
    TST.B     DrMenuFichier(A5)
    BEQ.S     @1
    LEA       TamponIODestination∫
              (A5),A0
    _Close
    BSR       InfoFichier
@1  LEA       TamponIOEntre(A5),A0
    _Close
    LEA       TamponIOSort(A5),A0
    _Close
    RTS

TmNumero
    MOVE.L    HandleMFichier(A5),∫
              -(SP)
    MOVE      #mNumero,-(SP)
    TST.B     DrNumero(A5)
    SEQ       DrNumero(A5)
    MOVE.B    DrNumero(A5),-(SP)
    _CheckItem
    TST.B     DrNumero(A5)
    BEQ.S     @1
    BSR       AfficheChaineNumero
@2  BRA       FinMenu
@1  PEA       RectEffaceNumero
    _EraseRect
    BRA.S     @2

FiltreFichiers
    LINK      A6,#0
    MOVEM.L   D0-D2/A0-A2,-(SP)
    MOVE.L    8(A6),A0
    TST.B     ioFlAttrib(A0)
    BMI.S     FiltreNon
    MOVE      ioFlUsrWds+fdFlags(A0),∫
              D0
    ANDI      #$5000,D0
    BNE.S     FiltreNon
    CLR       12(A6)
    BRA.S     FiltreOK
FiltreNon
    MOVE      #$FFFF,12(A6)
FiltreOK
    MOVEM.L   (SP)+,D0-D2/A0-A2
    UNLK      A6
    MOVE.L    (SP),4(SP)
    ADDQ.L    #4,SP
    RTS

CurseurAttente
    SUBQ.L    #4,SP
    _TickCount
    MOVE.L    CompteurTiming(A5),D0
    ADDQ.L    #2,D0
    CMP.L     (SP)+,D0
    BMI.S     ChangeCurseur
    RTS
ChangeCurseur
    TST.B     DrMenuFichier(A5)
    BEQ.S     ChangeCurseurBis
    MOVE      #68,D0
    MOVE      Numero_Curseur(A5),D4
    MULU      D4,D0
    LEA       Curseur1,A2
    PEA       0(A2,D0)
ChangeCurseur2
    _SetCursor
    SUBQ.L    #4,SP
    _TickCount
    MOVE.L    (SP)+,∫
              CompteurTiming(A5)
    ADDQ      #1,Numero_Curseur(A5)
    ANDI      #$0003,∫
              Numero_Curseur(A5)
    RTS

ChangeCurseurBis
    MOVE      Numero_Curseur(A5),D4
    ANDI      #1,D4
    BEQ.S     @1
    PEA       Curseur1bis
    BRA.S     ChangeCurseur2
@1  PEA       Curseur2bis
    BRA.S     ChangeCurseur2
```

Source 'T_Pom's/4.Asm'

```
Niveau0
    BSR       CurseurAttente
    BSR       LireCaractere
    TST.B     D5
    BNE.S     @1
@2  RTS
@1  CMPI.B    #$C,D5
    BEQ.S     Changement_Page
    TST.B     DrDepart(A5)
    BEQ.S     @2
    CMPI.B    #1,D5
    BEQ.S     Repere_1
    CMPI.B    #$1B,D5
    BEQ.S     Changement_Couleur
    CMPI.B    #$1F,D5
    BEQ       Changement_Position
    CMPI.B    #$10,D5
    BEQ       Numero_Telephone
    CMPI.B    #$60,D5
    BEQ       Fin_Ensemble_60
    CMPI.B    #$18,D5
    BEQ       Fin_Ensemble_18
    CMPI.B    #$16,D5
    BEQ       Caractere_ALC
    CMPI.B    #32,D5
    BMI.S     @2
    BRA       Dispatch

Repere_1
    BSR       LireCaractere
    TST.B     D5
    BNE.S     @1
@2  RTS
@1  MOVE.B    D5,D0
    ANDI.B    #$0F,D0
    BEQ.S     @2
    CMPI.B    #9,D0
    BGT       @2
    BSR       LireCaractere
    TST.B     D5
    BEQ.S     @2
    CMPI.B    #4,D5
    BNE.S     @2
    ST        DrRepere(A5)
    RTS

Changement_Page
    BSR       Zero_Chaines_Ville
    ST        DrDepart(A5)
    ST        Dr_Attente_Ville(A5)
    SF        Dr_Attente_Nom(A5)
    RTS

Changement_Couleur
    BSR       LireCaractere
    TST.B     D5
    BNE.S     @1
@3  RTS
@1  CMPI.B    #$36,D5
    BNE.S     @2
    BSR       LireCaractere
    BRA.S     @3
@2  CMPI.B    #$42,D5
    BEQ.S     @4
    CMPI.B    #$47,D5
    BNE.S     @5
@4  MOVE.B    D5,DrCouleur(A5)
    BSR       Ville_Departement
    BSR       Ville_Ville
    BSR       SuiteNom_Adresse
    RTS
@5  BSR       Departement_Ligne
    RTS

Changement_Position
    BSR       LireCaractere
    TST.B     D5
    BNE.S     @1
@2  RTS
@1  MOVE.B    D5,DrLigne(A5)
    ANDI.B    #%00111111,DrLigne(A5)
    BSR       LireCaractere
    TST.B     D5
    BEQ.S     @2
    MOVE.B    D5,DrColonne(A5)
    ANDI.B    #%00111111,∫
              DrColonne(A5)
    BSR       Nom_Prenom
    BSR       Adresse_Fin
    RTS

Numero_Telephone
    BSR       LireCaractere
    TST.B     D5
    BNE.S     @1
@2  RTS
@1  CMPI.B    #2,D5
    BNE.S     @3
    SF        Dr_Attente_Prenom(A5)
    ST        DrNumero_en_cours(A5)
    SF        C_Numero(A5)
    BRA.S     @2
@3  CMPI.B    #3,D5
    BNE.S     @2
    SF        DrNumero_en_cours(A5)
    ST        Dr_Attente_Nom_∫
              Suite(A5)
    LEA       C_Nom(A5),A0
    MOVE.B    (A0),D0
    EXT       D0
    CMPI.B    #' ',0(A0,D0)
    BEQ.S     @4
    CMPI.B    #127,D0
    BEQ.S     @4
    ADDQ.B    #1,(A0)
    MOVE.B    #' ',1(A0,D0)
@4  RTS

Fin_Ensemble_60
    BSR       LireCaractere
    TST.B     D5
    BNE.S     @1
@2  RTS
@1  CMPI.B    #$12,D5
    BNE.S     @2
    BSR       LireCaractere
    TST.B     DrAttente_Affichage_∫
              Ville(A5)
    BEQ.S     @3
    CMPI.B    #10,DrLigne(A5)
    BGE.S     @2
    TST.B     C_Ville(A5)
    BEQ.S     @2
    TST.B     C_Departement(A5)
    BEQ.S     @2
    BSR       Affichage_Ville
    SF        DrAttente_Affichage_∫
              Ville(A5)
    BRA.S     @4
```

```
@3 TST.B     Dr_Attente_Nom(A5)
   BNE.S     @2
   BSR       Affiche_Tous
   TST.B     DrMenuFichier(A5)
   BEQ.S     @4
   TST.B     Dr_Erreur(A5)
   BEQ.S     @4
   BSR       FermeDestination
@4 ST        Dr_Attente_Nom(A5)
   SF        C_Nom(A5)
   RTS

Fin_Ensemble_18
   TST.B     Dr_Attente_Nom(A5)
   BNE.S     @1
   BSR       Affiche_Tous
   TST.B     DrMenuFichier(A5)
   BEQ.S     @2
   TST.B     Dr_Erreur(A5)
   BEQ.S     @2
   BSR       FermeDestination
@2 ST        Dr_Attente_Nom(A5)
   SF        C_Nom(A5)
@1 RTS

Caractere_ALC
   BSR       LireCaractere
   LEA       Table1,A0
   MOVEQ     #16,D0
@1 CMP.B     (A0),D5
   BEQ.S     @2
   ADDQ.L    #2,A0
   DBRA      D0,@1
   BRA.S     Caractere_ALC_Accent
@2 MOVE.B    1(A0),D5
   BRA       Dispatch
Caractere_ALC_Accent
   CMPI.B    #$41,D5
   BNE.S     Caractere_ALC_Accent_2
   BSR       LireCaractere
   LEA       Table_Gra,A0
   MOVEQ     #2,D0
@1 CMP.B     (A0),D5
   BEQ.S     @2
   ADDQ.L    #2,A0
   DBRA      D0,@1
   RTS
@2 MOVE.B    1(A0),D5
   BRA.S     Dispatch
Caractere_ALC_Accent_2
   CMPI.B    #$42,D5
   BNE.S     Caractere_ALC_Accent_3
   BSR       LireCaractere
   CMPI.B    #'e',D5
   BEQ.S     @1
   RTS
@1 MOVE.B    #'é',D5
   BRA.S     Dispatch
Caractere_ALC_Accent_3
   CMPI.B    #$43,D5
   BNE.S     Caractere_ALC_Accent_4
   BSR       LireCaractere
   LEA       Table_Cir,A0
   MOVEQ     #4,D0
@1 CMP.B     (A0),D5
   BEQ.S     @2
   ADDQ.L    #2,A0
   DBRA      D0,@1
   RTS
@2 MOVE.B    1(A0),D5
   BRA.S     Dispatch
Caractere_ALC_Accent_4
   CMPI.B    #$48,D5
   BNE.S     Caractere_ALC_Accent_5
   BSR       LireCaractere
   LEA       Table_Tre,A0
   MOVEQ     #4,D0
@1 CMP.B     (A0),D5
   BEQ.S     @2
   ADDQ.L    #2,A0
   DBRA      D0,@1
   RTS
@2 MOVE.B    1(A0),D5
   BRA.S     Dispatch
Caractere_ALC_Accent_5
   CMPI.B    #$4B,D5
   BEQ.S     @2
   RTS
@2 BSR       LireCaractere
   CMPI.B    #'c',D5
   BEQ.S     @1
   RTS
@1 MOVE.B    #'ç',D5
   BRA.S     Dispatch

Dispatch
   CMPI.B    #2,DrLigne(A5)
   BMI.S     @2
   TST.B     Dr_Attente_Ville(A5)
   BEQ.S     R_Departement
   CMPI.B    #5,DrColonne(A5)
   BMI.S     @2
   CMPI.B    #$47,DrCouleur(A5)
   BNE.S     @2
   CMPI.B    #63,C_Ville(A5)
   BEQ.S     @2
   PEA       C_Ville(A5)
   BSR       Place_Caractere
@2 RTS

R_Departement
   TST.B     Dr_Attente_∫
             Departement(A5)
   BEQ.S     R_Confirme
   CMPI.B    #5,DrColonne(A5)
   BMI.S     @2
   CMPI.B    #$42,DrCouleur(A5)
   BNE.S     @2
   CMPI.B    #63,C_Departement(A5)
   BEQ.S     @2
   PEA       C_Departement(A5)
   BSR       Place_Caractere
@2 RTS

R_Confirme
   TST.B     DrAttente_Affichage_∫
             Ville(A5)
   BEQ.S     R_Nom
   RTS

R_Nom
   TST.B     Dr_Attente_Nom(A5)
   BEQ.S     R_Prenom
   CMPI.B    #3,DrLigne(A5)
   BMI.S     @1
   CMPI.B    #5,DrColonne(A5)
   BMI.S     @1
   CMPI.B    #$47,DrCouleur(A5)
   BNE.S     @1
   CMPI.B    #127,C_Nom(A5)
   BEQ.S     @1
   PEA       C_Nom(A5)
   BSR       Place_Caractere
@1 RTS

R_Prenom
   TST.B     Dr_Attente_Prenom(A5)
   BEQ.S     R_Numero
   CMPI.B    #3,DrLigne(A5)
   BMI.S     @1
   CMPI.B    #5,DrColonne(A5)
   BMI.S     @1
   CMPI.B    #$47,DrCouleur(A5)
   BNE.S     @1
   CMPI.B    #127,C_Prenom(A5)
   BEQ.S     @1
   PEA       C_Prenom(A5)
   BSR       Place_Caractere
@1 RTS

R_Numero
   TST.B     DrNumero_en_cours(A5)
   BEQ.S     R_Nom_Suite
   CMPI.B    #31,C_Numero(A5)
   BEQ.S     @1
   PEA       C_Numero(A5)
   BSR       Place_Caractere
@1 RTS

R_Nom_Suite
   TST.B     Dr_Attente_Nom_∫
             Suite(A5)
   BEQ.S     R_Adresse
   CMPI.B    #4,DrLigne(A5)
   BMI.S     @1
   CMPI.B    #5,DrColonne(A5)
   BMI.S     @1
   CMPI.B    #$47,DrCouleur(A5)
   BNE.S     @1
   CMPI.B    #127,C_Nom(A5)
   BEQ.S     @1
   PEA       C_Nom(A5)
   BSR       Place_Caractere
@1 RTS

R_Adresse
   TST.B     Dr_Attente_Adresse(A5)
   BEQ.S     @1
   CMPI.B    #4,DrLigne(A5)
   BMI.S     @1
   CMPI.B    #5,DrColonne(A5)
   BMI.S     @1
   CMPI.B    #$42,DrCouleur(A5)
   BNE.S     @1
   CMPI.B    #127,C_Adresse(A5)
   BEQ.S     @1
   PEA       C_Adresse(A5)
   BSR       Place_Caractere
@1 RTS
```

Fichier 'T_Pom's.R'

```
T_Pom's
APPLPfAm

Type PfAm = STR
 ,0
 Pom's - Filtre abonnés Minitel - Mai 1987

Type MENU
 ,1
\14
 ^1À propos de 'T_Pom's'...
 (-

Type MENU
 ,2
Fichier
 ^2Nouveau.../O
 (-
 Fermer/F
 (-
 ^3Numéros

Type MENU
 ,3
Édition
 Annuler/Z
 (-
 Couper/X
 Copier/C
 Coller/V
 (-
 Effacer

Type MENU
```

```
,4
Source
 ^4Fichier 'Pom's 27'...
 !\12^5Minitel

Type MENU
 ,5
Quitter
 ^8vers le 'Finder'/Q
 ^9vers une application.../A

Type ALRT

   ,128 (4)
82 72 178 440
128
5555

Type DITL
   ,128 (4)
2

Button
56 256 80 352
OK

StaticText
16 64 48 352
^0^1^2^3

Type STR
   ,1 (4)
Erreur d'entrée/sortie.

   ,2 (4)
Le catalogue est saturé.

   ,3 (4)
Le volume est saturé.

Type DLOG
 ,1 (4)

50 72 290 440
Visible  NoGoAway
1
0
1

Type DITL
 ,1 (4)
14

StaticText
0 0 248 368
\20

IconItem
24 24 56 56
257

IconItem
192 312 228 344
266

StaticText
12 72 28 352
T_Pom's — V 1.0/Mac

StaticText
36 72 52 352
© 1987 Pom's/Éditions MEV

StaticText
52 72 68 352
Jean-Luc Bazanegue et Christian Piard

StaticText
76 24 92 352
"T_Pom's" utilise l'annuaire électronique

StaticText
92 24 108 352
— ou un fichier créé par le programme
'Minitel'

StaticText
108 24 124 352
du numéro 27 de Pom's — pour constituer des

StaticText
124 24 140 352
fichiers d'adresses directement ∫
exploitables.

StaticText
148 24 164 352
Les 'sources' (MDS 68000) de la version 1.0 se

StaticText
164 24 180 352
trouvent dans le numéro 30 de "Pom's".

StaticText
196 24 212 296
Pom's/Éditions MEV — 12, rue d'Anjou —

StaticText
212 24 228 296
78000 Versailles — Tél. : (1) 39 51 24 43.

Type BNDL
 ,128(32)
 PfAm 0
 ICN#
 0 128
 FREF
 0 128

Type FREF
 ,128(32)
 APPL 0

TYPE ICN# = GNRL

* Icône application
 ,128(32)
2
7FFFFFFE 80000001 A0000005 9FFFFFF9
98000019 93FE6FC9 9000A009 97FD4FE9
90018009 973D3CE9 907FD609 96FEAB69
90FD4509 96FAA369 90FD4509 92FAA349
987D4619 9FFFFFF9 A0000005 BFFFFFFD
60000006 25555504 20000004 2AAAAAA4
20000004 25555504 20000004 2AFFEA74
20000004 3FFFFFFC 20018004 3FFFFFFC
*
7FFFFFFE FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
7FFFFFFE 3FFFFFFC 3FFFFFFC 3FFFFFFC
3FFFFFFC 3FFFFFFC 3FFFFFFC 3FFFFFFC
3FFFFFFC 3FFFFFFC 3FFFFFFC 3FFFFFFC

TYPE ICON = GNRL

* Icône à propos
 ,257 (4)
7FFFFFFE 80000001 A0000005 9FFFFFF9
98000019 93FE6FC9 9000A009 97FD4FE9
90018009 973D3CE9 907FD609 96FEAB69
90FD4509 96FAA369 90FD4509 92FAA349
987D4619 9FFFFFF9 A0000005 BFFFFFFD
60000006 25555504 20000004 2AAAAAA4
20000004 25555504 20000004 2AFFEA74
20000004 3FFFFFFC 20018004 3FFFFFFC

* Nouveau
 ,258 (4)
7FFFFFFE 80000001 A0000005 9FFFFFF9
98000019 90000009 93DEDBC9 9198D989
9198D989 91985189 919C2189 91985189
9198D989 9198D989 919ED989 90000009
98000019 9FFFFFF9 A0000005 BFFFFFFD
60000006 25555504 20000004 2AAAAAA4
20000004 25555504 20000004 2AFFEA74
20000004 3FFFFFFC 20018004 3FFFFFFC

* Fermer
 ,259 (4)
7FFFFFFE 80000001 9F87E1F9 A0481205
A6499265 A0481205 BFCFF3FD 9F87E1F9
80000001 9F87E1F9 A0481205 A6499265
A0481205 BFCFF3FD 9F87E1F9 80000001
9F87E1F9 A0481205 A6499265 A0481205
BFCFF3FD 9F87E1F9 80000001 9F87E1F9
A0481205 A6499265 A0481205 BFCFF3FD
9F87E1F9 80000001 7FFFFFFE 00000000

* Source Fichier 27
 ,260 (4)
0FFFFC00 08000600 09134D00 0AAAA480
0B12A440 0A000420 0A0007F0 0833C010
08484010 08108010 08210010 08410010
08790010 08000010 08000010 09FC0010
08000010 083DE010 08000010 097CF810
08000010 083F9F10 08000010 098F8F10
08000010 087EEF90 08000010 09F3E590
08000010 08000010 08000010 0FFFFFF0

* Source Minitel
 ,261 (4)
7FFFFFFE 80000001 A0000005 9FFFFFF9
98186199 9FFFFFF9 90000009 9FFFFFF9
904C0C29 9FFFFFF9 96318C09 9FFFFFF9
90411849 9FFFFFF9 90000009 9FFFFFF9
98C31919 9FFFFFF9 A0000005 BFFFFFFD
60000006 25555504 20000004 2AAAAAA4
20000004 25555504 20000004 2AFFEA74
20000004 3FFFFFFC 20018004 3FFFFFFC

* Finder
 ,264 (4)
0FFFFFF0 10000008 13FFFFC8 17FFFFE8
1668F468 16D5EAE8 1645E2E8 16D5EAE8
1655EAE8 17FFFFE8 17998A68 179494E8
17949CE8 17948E68 17949E68 16949E68
173988E8 17FFFFE8 13FFFFC8 10000008
10000008 10000008 100001C8 1200FFC8
10000008 10000008 10000008 0FFFFFF0
08000010 0A000050 08000010 07FFFFE0

* Application
 ,265 (4)
00010000 00028000 00044000 00082000
00129000 00254800 0048A400 00904200
01282100 02144080 040A8040 09450020
12A28010 24514008 4828BF04 84144082
4A2A8041 25453022 1283C814 095E7F0F
04A23007 02510007 01288007 00906007
00401FE7 0020021F 00100407 00080800
00041000 00022000 00014000 00008000

* Pom's
 ,266 (4)
00000000 00000050 00000080 00001100
000008AA 00000155 000000AA 00001401
00002A02 00004514 0000A280 00015140
000228A0 00051440 00028A00 00014560
0028A2A0 00545140 008A2980 00453D7C
00A2FFD6 1451FEAB 2A28FD45 4515FAA3
A20AFD45 5100FAA3 2A007D46 14007AAA
0A003D54 05001AA8 02800DD0 01000660

INCLUDE        T_Pom's.Code
```

TPom's : Minitel, Apple // et le 11

Christian Piard

Ce programme a pour objet la récupération des informations accessibles sur le Minitel par le 11, c'est-à-dire les noms, numéros de téléphone et adresse des abonnés figurant à l'annuaire électronique.

Le programme constitue sur disque un fichier comprenant l'ensemble des abonnés affichés par le service, fichier imprimable, récupérable par un traitement de textes ou par votre propre programme de mailings par exemple.

Le programme ne fonctionne que sous ProDOS.

Configuration

Pour utiliser ce programme, vous devez disposer des éléments suivants :

Vous avez un Apple][+ :

- une ROM pour l'affichage des minuscules ;
- un Minitel,
- une carte 'langage' qui accueillera ProDOS ;
- une carte SuperSérie Apple ;
- un câble de liaison type Apple //e.

Vous avez un Apple //e équipé ou non du 65C02 :

- un Minitel ;
- une carte SuperSérie Apple ;
- un câble de liaison type Apple //e.

Vous avez un Apple //c :

- un Minitel ;
- une carte SuperSérie Apple ;
- un câble de liaison type Apple //c.

Vous avez un Apple IIGS :

- un Minitel,
- un câble de liaison type Macintosh Plus (on utilise le port série intégré). Si vous désirez utiliser une carte SuperSérie installée dans l'un des slots, il faut un câble type Apple //e.

Tous les types de Minitel conviennent. Le câble peut être réalisé d'après les schémas des numéro 27 et 28 de Pom's ou commandé à la revue (bon de commande page 74).
Le programme se charge de rechercher la carte SSC pour le port série en commençant par le port 7 pour finir par le 1. Une carte en port 2 sera donc utilisée même s'il en existe une également en port 1 (cas du //c et du IIGS).

Mode d'emploi

Le programme crée sur disque un fichier de type TEXT comprenant les abonnés consultés à l'aide du Minitel. Les différents options du menu sont les suivantes :

Recevoir l'annuaire avec numéros de téléphone

Option R : Le programme vous demande le nom du fichier TEXT à créer avant de se mettre à l'écoute du Minitel.

Le nom de fichier doit respecter la syntaxe ProDOS (la routine de saisie élimine la plupart des fautes possibles). Il peut être de la forme :

`/RAM/CHEMINEES78`
ou
`CHEMINEES`

Dans ce dernier cas, le préfixe par défaut est utilisé pour déterminer le volume d'accueil du fichier.

Si le fichier existe déjà sur le volume, le programme refusera de l'écraser, ceci pour éviter les conséquence d'une fausse manœuvre.

Dans le cas où vous désirez réellement remplacer un fichier existant appuyer sur :

- ⌘ (IIGS) ou
- Option (IIGS) ou
- (//e, //c) ou
- (//e, //c) ou
- un bouton du joystick (][+)

lorsque vous faites le `RETURN` de validation du nom de fichier. L'éventuel ancien fichier sera écrasé.

Le programme affiche alors `fichier TEXT créé`. Il ne reste plus qu'à commencer la consultation sur Minitel. La mémoire disponible pour les adresses est d'environ 26000 octets, ce qui suffit pour 3 à 400 adresses. Le compteur affiché à l'écran vous indique en permanence la place disponible.

À tout moment, pour libérer de la place en mémoire, il est possible de presser `S` ce qui provoque l'écriture sur le volume ; la taille du fichier n'est donc limité que par la place disponible sur le disque. En fin de consultation, presser `ESC` ce qui sauvegarde la mémoire et reconduit au menu général.
Au fur et à mesure de l'enregistrement, les noms, adresses et numéros de téléphone apparaissent à l'écran.

Recevoir l'annuaire sans numéros de téléphone

Option L : Toutes les indications de l'option précédente restent valables. Seule différence : les numéros de téléphone ne feront pas partie du fichier : c'est donc la solution à retenir dans la perspective d'un mailing.

Imprimer un fichier

Option I : Conçue comme accessoire, cette option permet d'imprimer un fichier TEXT sans devoir nécessairement charger

AppleWorks, AppleWriter ou autre.

Un fichier d'adresses vient d'être créé : cette option permet d'obtenir rapidement sa représentation papier, sans souci de présentation.

Le programme, qui suppose l'imprimante connectée au port 1, interrompra l'impression si vous pressez ***ESC***.

L'impression de fichier d'un type autre que TEXT sera refusée ; toutefois, les fichiers de textes AppleWorks sont admis.

Modifier le préfixe

Option M : C'est l'option à choisir pour changer le préfixe par défaut, préfixe qui est rappelé en haut de l'écran. Par exemple :

/UNIDISK2/POMS30/T3

Volumes en ligne

Option V : Option indispensable pour identifier les disquettes, disques virtuels et autres volumes actuellement disponibles.

Catalogue

Option C : Permet le listage des fichiers du catalogue visé par le préfixe par défaut. Si celui-ci est :

/PROFILE/CLIENT

seuls les fichiers du dossier CLIENT apparaîtront.

Quitter

Option Q : Contrairement à *CTRL-RESET*, cette option permet de quitter le programme.

Le filtrage

La récupération des éléments purement TEXT de l'annuaire électronique présente bien des difficultés du fait du caractère non standard de la présentation. Il faudra œuvrer en traitement de textes avant d'envisager l'impression d'étiquettes-adresse (ne serait-ce que pour ajouter les code postaux indisponibles dans la consultation).

Pour l'identification de ce qui doit être un nom, une adresse ou un téléphone, le programme se base sur la couleur et la position des éléments mais le filtrage et la reconnaissance des divers éléments n'est pas efficace à 100%. Les mentions du type *futur numéro* ou *téléphone de voiture* en lieu et place de l'adresse sont difficilement décelables. Le programme ne donnera pas un résultat satisfaisant lorsque plusieurs numéros sont affectés à un abonné. Selon la présentation – très variable – la récupération se fera plus ou moins bien.
Les copies d'écran Minitel annexées illustrent ces variations.

Pour une page affichée sur l'écran du Minitel, 600 à 1000 caractères (texte, contrôle et graphisme) sont émis sur la prise péri-informatique. Après filtrage des caractères inutiles pour nous, il peut ne rester que 4 ou 5 adresses de 70 caractères soit 600 ou 700 caractères éliminés.

Précautions, Remarques, Mailing

Le programme, pour constituer des adresses cohérentes, attend un effacement de page toujours suivi de l'affichage de la ville en haut de l'écran.

Il faut donc mettre le programme à l'écoute du Minitel avant même de commencer la consulation pour que l'Apple voit bien passer la ville. Sinon, rien ne sera enregistré.

Dans le fichier TEXT, les adresses sont séparées par *** pour que votre programme ou votre traitement de texte identifie facilement le début et la fin.
Pour réaliser un mailing à l'aide d'AppleWriter par exemple, remplacer les *** par des sauts de page par la commande :

CTRL-B

*CTRL-F /***/.FF/A↵*

Spécifier un espace entre pages de 10 lignes si vos étiquettes sont distantes de 10 lignes :

CTRL-P IP10↵

La liaison Minitel/Apple peut être vérifiée de la façon suivante : programme prêt à enregistrer, presser des touches sur le clavier du Minitel. À chaque caractère reçu par l'Apple, le premier '–' de la ligne 8 doit se transformer en un '+' puis en un '–', etc.

Deux pages de l'annuaire illustrant la 'variété' de la présentation

```
*****************************************
   FNAC
   PARIS PARIS                        1
-----------------------------------------
> 1 F.N.A.C. (Fédération
   Nationale d'Achats des
   Cadres)
-----------------------------------------
   F.N.A.C Autoradio
   (Fédération
   Nationale d'Achat
   des Cadres)

  2 -54  av Bosquet  7e  (1) 45 51 28 15

  3 -15Ter  bd Gouvion   (1) 45 74 57 41
    St Cyr  17e

  4 -53  r Richard       (1) 43 67 38 05
    Lenoir  11e

-----------------------------------------
> plus d'informations
  tapez le N° choisi:... puis >  ENVOI
                 page suivante >  SUITE
*****************************************
```

```
*****************************************
   MINISTERE
   PARIS PARIS                        1
-----------------------------------------
> 1 Ministère Agriculture
-----------------------------------------
  2 Ministère Défense     (1) 42 61 57 31
    231  bd St Germain  7e
-----------------------------------------
  3 Ministère Justice     (1) 42 60 13 60
    13  pl Vendome  1e
-----------------------------------------
    Ministère Justice

  4 -13  pl Vendome  1e  (1) 42 61 80 22

  5 -meme adresse        (1) 42 61 55 85

-----------------------------------------
> plus d'informations
  tapez le N° choisi:... puis >  ENVOI
                 page suivante >  SUITE
*****************************************
```

Source T.POMS.1
Assembleur ProCODE

```
PUT     T.POMS.2
PUT     T.POMS.3
PUT     T.POMS.4
```

Représentation d'une page de l'annuaire électronique (633 caractères)

```
2600:14 0C 00 00 1E 1F 41 45 1B 47 4D 49 4E 49 53 54  -......AE.GMINIST
2610:45 52 45 1F 42 45 1B 47 50 41 52 49 53 20 1B 42  -ERE.BE.GPARIS .B
2620:50 41 52 49 53 20 1F 43 41 1B 44 60 12 67 1F 41  -PARIS .CA.D`.g.A
2630:41 1B 46 2E 12 42 14 1F 30 34 18 00 00 00 0A 18  -A.F..B..04......
2640:00 00 00 0A 18 00 00 00 0A 18 00 00 00 0A 18 00  -................
2650:00 00 0A 18 00 00 00 0A 18 00 00 00 0A 18 00 00  -................
2660:00 0A 18 00 00 00 0A 18 00 00 00 0A 18 00 00 00  -................
2670:0A 18 00 00 00 0A 18 00 00 00 0A 18 00 00 00 0A  -................
2680:18 00 00 00 0A 18 00 00 00 0A 18 00 00 00 0A 18  -................
2690:00 00 00 0A 18 00 00 00 0A 18 00 00 00 0A 18 14  -................
26A0:1F 42 66 1B 42 20 20 35 1F 44 43 1B 47 01 31 04  -.Bf.B  5.DC.G.1.
26B0:20 1F 44 45 1B 47 4D 69 6E 69 73 74 16 41 65 72  - .DE.GMinist.Aer
26C0:65 20 64 65 20 6C 61 20 1F 44 5A 1B 47 10 02 28  -e de la .DZ.G..(
26D0:31 29 20 34 37 20 30 33 20 34 39 20 38 38 10 03  -1) 47 03 49 88..
26E0:1F 45 45 1B 47 4A 75 73 74 69 63 65 1F 46 45 1B  -.EE.GJustice.FE.
26F0:42 31 33 20 20 70 6C 20 56 65 6E 64 16 43 6F 6D  -B13  pl Vend.Com
2700:65 20 20 31 65 1F 47 45 1B 44 60 12 63 1F 48 43  -e  1e.GE.D`.c.HC
2710:1B 47 01 32 04 20 1F 48 45 1B 47 4D 69 6E 69 73  -.G.2. .HE.GMinis
2720:74 16 41 65 72 65 20 64 65 20 6C 61 20 4D 65 72  -t.Aere de la Mer
2730:1F 48 5A 1B 47 10 02 28 31 29 20 34 32 20 37 33  -.HZ.G..(1) 42 73
2740:20 35 35 20 30 35 10 03 1F 49 45 1B 42 33 20 20  - 55 05...IE.B3
2750:70 6C 20 46 6F 6E 74 65 6E 6F 79 20 20 37 65 1F  -pl Fontenoy  7e.
2760:4A 45 1B 44 60 12 63 1F 4B 43 1B 47 01 33 04 20  -JE.D`.c.KC.G.3.
2770:1F 4B 45 1B 47 4D 69 6E 69 73 74 16 41 65 72 65  -.KE.GMinist.Aere
2780:20 64 65 20 6C 61 20 1F 4B 5A 1B 47 10 02 28 31  - de la .KZ.G..(1
2790:29 20 34 35 20 36 37 20 35 35 20 34 34 10 03 1F  -) 45 67 55 44...
27A0:4C 45 1B 47 53 61 6E 74 16 42 65 1F 4D 45 1B 42  -LE.GSant.Be.ME.B
27B0:31 20 20 70 6C 20 46 6F 6E 74 65 6E 6F 79 20 20  -1  pl Fontenoy
27C0:37 65 1F 4E 45 1B 44 60 12 63 1F 4F 45 1B 47 4D  -7e.NE.D`.c.OE.GM
27D0:69 6E 69 73 74 16 41 65 72 65 20 64 65 73 20 1F  -inist.Aere des .
27E0:50 45 1B 47 52 65 6C 61 74 69 6F 6E 73 20 45 78  -PE.GRelations Ex
27F0:74 16 42 65 72 69 65 75 72 65 73 1F 52 43 1B 47  -t.Berieures.RC.G
2800:01 34 04 20 2D 1F 52 46 1B 42 33 37 20 20 71 75  -.4. -.RF.B37  qu
2810:61 69 20 4F 72 73 61 79 20 20 37 65 1F 52 5A 1B  -ai Orsay  7e.RZ.
2820:47 10 02 28 31 29 20 34 35 20 35 35 20 39 35 20  -G..(1) 45 55 95
2830:34 30 10 03 14 1F 32 33 0D 18 00 00 1F 32 33 1B  -40....23.....23.
2840:44 60 12 67 14 1F 58 51 0D 18 00 00 1F 58 51 1B  -D`.g..XQ.....XQ.
2850:42 70 61 67 65 0E 20 0F 73 75 69 76 61 6E 74 65  -Bpage. .suivante
2860:0E 20 0F 16 2E 1B 5A 1B 50 20 1B 5D 20 53 55 49  -. ....Z.P .] SUI
2870:54 45 20 20 14 1F 41 41 11                       -TE  ..AA..
```

ProDOS

][+ //e //e+ //c][gs

Récupération par T_Pom's sans n° de téléphone (206 caractères)

Seuls les caractères TEXT utilisables sont gardés ; les caractères accentués sont recodés avant stockage.

```
Ministère de la
Justice
13  pl Vendome  1e
PARIS
***
Ministère de la Mer
3  pl Fontenoy  7e
PARIS
***
Ministère de la
Santé
1  pl Fontenoy  7e
PARIS
***
Relations Extérieures
37  quai Orsay  7e
PARIS
***
```

Source T.POMS.2
Assembleur ProCODE

```
*----------------
* ANNUAIRE
* (c) Pom's & CP
*----------------
*120587
        DSK  T.POMS
        ORG  $2000

COMPT    =  $18
FILEBUF  =  $1000
BUFONLIN =  $1500
BUFFIN   =  $1600
PATHNAME =  $1700
MLI      =  $BF00
CREATE   =  $C0
OPEN     =  $C8
READ     =  $CA
WRITE    =  $CB
CLOSE    =  $CC
GETFINFO =  $C4
ONLINE   =  $C5
STPREFIX =  $C6
GTPREFIX =  $C7
GETTIME  =  $82
GETEOF   =  $D1
SETEOF   =  $D0
KBD      =  $C000
STROBE   =  $C010
HP       =  $C030
COUT     =  $FDED
GBASL    =  $26
GBASCALC =  $F847
VTAB     =  $FC22
CV       =  $25
WNDTOP   =  $22
WNDBTM   =  $23
CH       =  $24
STROUT   =  $DB3A
HOME     =  $FC58
TEXT     =  $FB39
BASIC    =  $3D0
CLREOP   =  $FC42
CLREOL   =  $FC9C
LINPTR   =  $ED24
LASTDEV  =  $BF30

DEB00   LDX  #$FF
        TXS
        LDA  $BF00
        CMP  #$4C
        BEQ  DEV0
        LDA  #$7F
        JSR  ERREUR
        JMP  BASIC

DEV0    JSR  VERSSC
        BCC  DEV00
        JSR  HOME
        LDY  #0
]1      LDA  PASSSC,Y
        BEQ  ]2
        JSR  COUT
        INY
        BNE  ]1
        JSR  BIP
]2      JMP  BASIC

DEV00   LDA  $FBB3
        CMP  #6
        BNE  DEV0E
        LDA  $FBC0
        BNE  ]1
        JSR  DEUXC

]1      LDA  INT
        JSR  $FE95
        LDA  #7
        JSR  COUT
        JSR  $FE93
        LDA  #7
        JSR  SSC
        JSR  PARASSC
        JSR  INITSSC
```

```
DEVOE   LDA  #<RESET
        STA  $3F2
        LDA  #>RESET
        STA  $3F3
        EOR  #$A5
        STA  $3F4

        LDA  LASTDEV
        STA  LD
        JSR  MLI
        DFB  ONLINE
        DA   PARLD0
        BCS  MENU
        LDA  PATHNAME
        AND  #%00001111
        STA  PATHNAME
        TAY
]1      LDA  PATHNAME,Y
        INY
        STA  PATHNAME,Y
        DEY
        DEY
        CPY  #0
        BNE  ]1
        LDA  #"/"
        STA  PATHNAME+1
        INC  PATHNAME

        JSR  MLI
        DFB  STPREFIX
        DA   PARLD1

        JMP  MENU

*----------------
* MENU
*----------------

ADRCHOIX DA  RECANN
         DA  RECANNST
         DA  IMPRIME
         DA  MODPREF
         DA  ONL
         DA  CATAL
         DA  QUITTER

MENU     LDA  #21
         JSR  COUT
         JSR  TEXT
         JSR  HOME
         JSR  CLOSEFIL

         LDA  #1
         JSR  LIGNE
         LDA  #22
         JSR  LIGNE
         LDX  #0
]1       LDA  MESTITRE,X
         BEQ  ]2
         JSR  COUT
         INX
         BNE  ]1
]2       JSR  LITPREF
         LDA  #3
         STA  WNDTOP
         LDA  #22
         STA  WNDBTM
         LDA  #4
         JSR  VTAB1
         LDY  #8
]3       LDA  POMS,Y
         STA  $49F,Y
         DEY
         BPL  ]3

CHOIX    LDA  #4
         JSR  VTAB1
         LDY  #0
]1       LDA  MESMENU,Y
         BEQ  ]2
         JSR  COUT
         INY
         BNE  ]1
]2       LDA  #1
         STA  LONG
         LDA  #19
         JSR  GBASCALC
         LDA  #35
         ADC  GBASL
         STA  GBASL
         BCC  ]3
         INC  GBASL+1

]3       JSR  INPUT
         LDY  #6
         LDA  BUFFIN
]4       CMP  MESCHOIX,Y
         BEQ  ]5
         DEY
         BPL  ]4
         JSR  BIP
         BCS  ]2

]5       LDA  #<ADRCHOIX
         STA  CHOIXIND+1
         TYA
         BEQ  CHOIXIND
]6       INC  CHOIXIND+1
         INC  CHOIXIND+1
         DEY
         BNE  ]6

CHOIXIND JMP  (ADRCHOIX)

LIGNE    JSR  GBASCALC
         LDY  #39
         LDA  #"_"
]1       STA  (GBASL),Y
         DEY
         BPL  ]1
         RTS
```

```
*----------------
* RECEVOIR ANNUAIRE
*----------------

RECANNST LDA  #0          ; réception sans n° téléphone
         STA  TELEPH
         BEQ  RECANN1

RECANN   LDA  #1          ; réception avec n°
         STA  TELEPH
RECANN1  JSR  HOME
         JSR  LITPREF

         LDA  LNGPATHN    ; cherche place disponible sur
         STA  PATHNAME    ; le volume
         LDA  #4
         JSR  VTAB1
         JSR  GFI
         BCC  ]1
         JSR  GETRET
         JMP  MENU

]1       LDA  #8
         STA  CH
         LDA  AUXTYPG     ; l'affiche
         SEC
         SBC  BLOCKUS
         TAX
         LDA  AUXTYPG+1
         SBC  BLOCKUS+1
         JSR  LINPTR
         INC  CH

         LDX  #0
]2       LDA  BL,X
         BEQ  ]3
         JSR  COUT
         INX
         BNE  ]2
]3       LDA  #6          ; limite fenetre
         STA  WNDTOP

         LDY  #39         ; affiche message en bas
]30      LDA  BAS,Y       ; de l'écran
         STA  $7D0,Y
         DEY
         BPL  ]30

]3000    JSR  SAISIN1     ; saisie du nom de fichier
         BCC  ]300
         JMP  MENU
]300     JSR  CREFIC      ; création du fichier
         BCS  RECANN1
]301     LDA  LNGPATHN
         STA  PATHNAME
         JSR  OUVREFIC    ; ouverture du fichier
         BCC  ]31
         JSR  ERREUR
         JSR  GETRET
         JMP  MENU
]31      JSR  VIDEF       ; remet le fichier à
         BCC  ]31A        ; 0 s'il existait
         JSR  GETRET
         JMP  MENU
]31A     JSR  HOME
         JSR  INICOMPT
         JSR  TRAIT1      ; affiche mémoire disponible

         LDA  #13
         JSR  VTAB1

         LDY  #0          ; affiche 'prêt à recevoir'
]310     LDA  PRET,Y
         BEQ  ]311
         JSR  COUT
         INY
         BNE  ]310

]311     LDA  #10
```

```
        JSR  VTAB1
DR      JSR  INICOMPT   ; initialise compteur de
        LDA  #$42       ; ville présumée en 2ème ligne
        STA  HVILLE
        LDA  #$FF       ; aucun caractère reçu
        STA  DP
BOUCLE  JSR  R1C        ; boucle principale : on teste
        CMP  #$1F       ; les caractères de contrôle de
        BEQ  R1F        ; l'affichage Minitel et on
                        ; positionne
        CMP  #$1B       ; des drapeaux en conséquence
        BNE  ]10        ; pour afficher et stocker ou
        JMP  R1B        ; non les carac qui suivront
]10     CMP  #$10
        BNE  ]11
        JMP  R10
]11     CMP  #$18
        BNE  ]111
        JMP  R18
]111    CMP  #$C
        BNE  ]2
        JMP  RC
]2      CMP  #$60
        BNE  ]3
        JMP  R60
]3      CMP  #$16
        BNE  ]4
        JMP  R16
]4      LDX  VR         ; si on n'a pas reçu la ville
        BEQ  BOUCLE     ; on n'affiche rien.
        JMP  RAUTRE     ; affiche éventuellement
R1F     LDA  N          ; reçu $1F
        BEQ  ]2         ; si nom en cours, fin du nom
        JSR  FINNOM
]2      LDA  A
        BEQ  ]3         ; si adress en cours, fin adress
        JSR  FINADR
]3      LDA  P          ; si prénom en cours, fin prénom
        BEQ  ]30
        JSR  FINPRE
]30     JSR  R1C        ; recevoir 2 caractères de plus
        STA  F1         ; F1 et F2, qui suivent $1F
        JSR  R1C
        STA  F2

        LDA  F1         ; si F1 = 42 (ou 43…) et F2 = 45
        CMP  HVILLE     ; on va recevoir la ville.
        BCS  ]300       ; < 42, on boucle
        JMP  BOUCLE
]300    BNE  ]5
        LDA  F2
        CMP  #$45
        BNE  ]4
        LDA  #1
        STA  V          ; c'est la ville
        LDA  #0
        STA  PP
        JSR  R1C        ; évite le premier $1B $47
        JSR  R1C        ; avant de recevoir la ville
        JMP  RVILLE
]4      JMP  BOUCLE

]5      LDA  F2         ; a <> $42, ce n'est pas la vill
        CMP  #$45       ; F2 = 45 ?
        BEQ  ]6         ; oui, on recevra nom ou adresse
        BCS  ]7         ; > à 45, on recevra le prénom
        JMP  BOUCLE     ; autre chose, on boucle

]6      LDA  F1         ; nom et adresse ne peuvent être
        CMP  #$44       ; avant ligne $45
        BCC  ]60
        LDA  #1         ; on recevra nom ou adresse
        STA  NA
        LDA  #0
        STA  PP
]60     JMP  BOUCLE

]7      LDA  #1         ; on recevra prénom
        STA  P
        JMP  BOUCLE
R1B     LDA  NA         ; on hésitait entre nom/adress ?
        BNE  ]22
        JSR  R1C
        JMP  BOUCLE     ; non, on boucle
]22     JSR  R1C        ; oui, on reçoit un autre caract
        CMP  #$42       ; c'est 42, on recevra peut-être
        BNE  ]3         ; une adresse
        LDA  AP         ; adresse possible ?
        BEQ  ]220
        LDA  #1
        STA  A
        LDA  #0
        STA  PP
        STA  NA
]220    JMP  BOUCLE
]3      CMP  #$47       ; c'est 47, on recevra un nom
        BNE  ]4
        LDA  #1
        STA  N
        LDA  #0
        STA  PP
        STA  NA
]4      JMP  BOUCLE

R10     JSR  R1C        ; reçu 10, on attend un caract
        CMP  #2         ; c'est 2, alors début du téléph
        BNE  ]1
        LDA  #1
        STA  T
        LDA  #0
        STA  PP
]1      CMP  #3         ; c'est 3, alors fin du téléph
        BNE  ]2
        LDA  #0
        STA  T
        STA  PP
]2      JMP  BOUCLE     ; autre, on boucle
R18     LDA  #0         ; reçu $18 ou $60 $12
        STA  V          ; fin d'un groupe ou d'une page
        STA  N          ; réinitialise les drapeaux
        STA  A
        STA  NA
        STA  T
        STA  P
        STA  AP
        STA  PP
        BIT  DP
        BMI  ]1
        JSR  AFVILLE
        JSR  TRAIT
]1      JMP  BOUCLE

R60     JSR  R1C        ; reçu $60, attend 1 caract
        CMP  #$12       ; si c'est 12, fin du groupe
        BNE  ]1         ; nom + adresse + tél + comment
        JMP  R18
]1      JMP  BOUCLE     ; sinon on boucle
RC      LDA  #0
        STA  VR         ; ville non reçue
        JMP  R18

R16     JSR  R1C        ; reçu $16, alphabet G2
        TAX
        AND  #$F0       ; si carac = $4x, c'est une
        CMP  #$40       ; minuscule accentuée ou ç
        BEQ  R162

        TXA             ; ce n'est pas un accent
        LDY  #0
]1      CMP  ACT1,Y     ; cherche dans la table
        BEQ  ]2
        INY
        INY

        CPY  #34
        BNE  ]1
        JMP  BOUCLE
]2      INY
```

```
        LDA  ACT1,Y     ; charge carac correspondant
        JMP  RAUTRE     ; avant traitement normal

R162    STX  C1         ; on attend le deuxième
        JSR  R1C        ; caractère
        STA  C2

        LDY  #0
]1      LDA  C1
        CMP  ACT2,Y     ; compare le premier
        BNE  ]2
        INY
        LDA  C2
        CMP  ACT2,Y     ; compare le deuxième
        BNE  ]3
        INY
        LDA  ACT2,Y
        JMP  RAUTRE
]2      INY
]3      INY
        INY
        CPY  #42
        BNE  ]1
        JMP  BOUCLE

RAUTRE  CMP  #$20
        BCC  ]1
        STA  CRECU
        LDA  TELEPH
        AND  T          ; opérations logiques sur
        STA  T          ; les drapeaux pour décider
        LDA  P          ; si on affichera ou non
        AND  PP         ; le caractère reçu.
        ORA  N
        ORA  A
        ORA  T
        ORA  NA
        BEQ  ]1
        LDA  CRECU
        STA  CRECU1
        AND  #$7F
        STA  DP
        JSR  COUT1
]1      JMP  BOUCLE

FINNOM  BIT  DP
        BMI  ]1
        JSR  CROUT
]1      LDA  #0
        STA  N
        LDA  #1
        STA  PP         ; prénom possible
        STA  AP         ; adresse possible
        RTS

FINADR  BIT  DP
        BMI  ]1
        JSR  CROUT
]1      LDA  #0
        STA  A
        RTS

FINPRE  BIT  DP         ; si on n'a pas reçu de caract
        BMI  ]1         ; il n'y a pas de prénom
        JSR  CROUT
]1      LDA  #0
        STA  P
        RTS

RVILLE  LDA  #0         ; stocke ville
        STA  VILLE
]4      JSR  R1C
        CMP  #$1F
        BEQ  ]6         ; ce n'était pas la ville
        CMP  #$1B
        BEQ  ]5
        INC  VILLE
        LDY  VILLE
        STA  VILLE,Y
        CPY  #30
        BNE  ]4
]5      LDA  #0
        STA  V
        LDA  #$42
        STA  HVILLE
        LDA  #1
        STA  VR         ; ville reçue
        JMP  BOUCLE
]6      LDA  #$43       ; ville en ligne 3 et non 2
        STA  HVILLE     ; car la rubrique occupait 2
        JMP  R1F        ; lignes

COUT1   ORA  #$80       ; 'poidsforise' le caractère
        JSR  STOCK      ; stockage et affichage
        JMP  COUT

AFVILLE LDA  VILLE      ; affiche et stocke la ville
        BNE  ]2
        RTS
]2      LDY  #1
]1      LDA  VILLE,Y
        JSR  COUT1
        INY
        CPY  VILLE
        BNE  ]1
        JMP  CROUT

R1C     JSR  RECOIT     ; reçoit 1 caractère
        BCS  ]1         ; retenue = 0 : pas reçu
        LDA  KBD        ; une touche pressée ?
        BIT  STROBE
        AND  #%11011111 ; convertit en majuscule
        CMP  #"S"       ; si S, on sauvegarde
        BEQ  SVM        ; et on continue à enregistrer
        CMP  #$9B       ; si ESC, on sauvegarde et
        BNE  R1C        ; on termine
        JSR  SAUVEMEM
        LDX  #$FF
        TXS
        JMP  MENU
]1      RTS
SVM     JSR  SAUVEMEM
        LDX  #$FF
        TXS
        JMP  DR

TRAIT   BIT  DP         ; a-t-on reçu des caractères ?
        BMI  TRAIT2
        JSR  CROUT
        JSR  $FC42      ; efface fin écran
        LDA  #$8D
        JSR  STOCK
        LDY  #3         ; met un séparateur entre deux
]1      LDA  #"*"       ; adresses
        JSR  STOCK
        DEY
        BNE  ]1
        LDA  #$8D
        JSR  STOCK
        LDY  #$FF
        STY  DP

TRAIT1  LDA  #8
        JSR  VTAB1
        LDA  #0
        STA  CH

        LDY  #25        ; affiche mémoire disponible
]100    LDA  #"-"
        JSR  COUT
        DEY
        BPL  ]100
        LDA  #"/"
        JSR  COUT
        LDA  #" "
        JSR  COUT
        SEC
        LDA  #$FF
        SBC  COMPT
        TAX
        LDA  #$95
        SBC  COMPT+1
        JSR  LINPTR
        LDA  #" "
        JSR  COUT
```

```
        LDA  #"/"
        JSR  COUT
]101    LDA  #"-"
        JSR  COUT
        LDA  CH
        BNE  ]101
        JSR  CROUT
        LDA  #10
        JMP  VTAB1
TRAIT2  JMP  CROUT
STOCK   STA  CARAC       ; stocke les caractères reçus
        STY  RY          ; et filtrés

        LDA  COMPT
        CMP  #$FF        ; mémoire saturée ?
        BNE  ]1
        LDA  COMPT+1
        CMP  #$95
        BEQ  ]FIN        ;oui, on enregistre
]1      INC  COMPT
        BNE  ]2
        INC  COMPT+1

]2      LDA  CARAC
        LDY  #0
        STA  (COMPT),Y
        LDY  RY
        RTS
]FIN    JMP  SVM

CROUT   LDA  CRECU1
        CMP  #$8D
        BNE  ]1
        RTS
]1      LDA  #$8D        ; stocke un retour chariot
        STA  CRECU1
        JSR  STOCK
        JSR  CLREOL      ; efface fin de la ligne
        JMP  $DAFB       ; crout

SAUVEMEM LDA #$8D
        STA  $3000
        SEC
        LDA  COMPT+1
        SBC  #$30
        STA  LAS+1
        LDA  COMPT
        STA  LAS
        JSR  EF
        BCC  ]1
        JSR  ERREUR
        JSR  GETRET
]1      RTS

INICOMPT LDA #0
        STA  COMPT
        LDA  #$30
        STA  COMPT+1
        RTS


VILLE   DS   40
F1      DS   1
F2      DS   1
V       DS   1
P       DS   1
NA      DS   1
AP      DS   1
A       DS   1
T       DS   1
N       DS   1
CRECU   DS   1
CRECU1  DS   1
DP      DS   1
HVILLE  DS   1
VR      DS   1
TELEPH  DS   1
RY      DS   1
CARAC   DS   1
C1      DS   1
C2      DS   1
BL      ASC  "blocs libres sur ce disque"
        DFB  0
```

```
BAS     ASC  "<ESC> Sauvegarde/Fin      <S> Sauvegarde
PRET    ASC  "            Fichier TEXT créé            "
        ASC  "  Pret pour consultation de l'annuaire  "
        DFB  0

*---------------
* IMPRIMER UN FICHIER
*---------------

IMPRIME JSR  HOME
        JSR  LITPREF

]1      JSR  SAISIN2     ; saisie du nom de fichier
        BCC  ]2
        JMP  MENU
]2      LDA  LNGPATHN
        STA  PATHNAME
        JSR  OUVREFIC    ; ouverture du fichier
        BCC  ]3
]20     JSR  ERREUR
        JSR  GETRET
        JMP  IMPRIME

]3      JSR  GFI         ; get file info pour type de
        BCS  ]20         ; fichier

        LDA  FILETYG
        CMP  #4          ; fichier TEXT ?
        BEQ  ]3B
        CMP  #$1A        ; fichier de texte AppleWorks ?
        BEQ  ]3B
        LDA  #$7C        ; non, erreur
        BNE  ]20

]3B     LDA  #1          ; ne lire qu'un caractère
        STA  LAS
        LDA  #0
        STA  LAS+1

]30     JSR  LITFIC      ; le lire
        BCC  ]4
        CMP  #$4C        ; fin de fichier ?
        BEQ  FINIMPRI
        JSR  ERREUR
        JSR  GETRET
        JMP  MENU

]4      LDA  $3000
        ORA  #$80
        CMP  #$8D        ; est-ce un CR
        BNE  ]5

        LDY  #79         ; on n'imprimera pas plus de
        STY  NBCR        ; 80 caractères par ligne.
]5      JSR  $C100       ; (utile si ce n'est pas un
        DEC  NBCR        ; fichier d'adresse qui est
        BPL  ]6          ; imprimé)
        LDA  #$8D
        JSR  $C100
        LDY  #79
        STY  NBCR
]6      LDA  KBD
        BIT  STROBE
        CMP  #$9B
        BNE  ]30
FINIMPRI LDA #$8D
        JSR  $C100
        JSR  $FE93
        JMP  MENU

NBCR    DS   1
```

Source T.POMS.3
Assembleur ProCODE

```
*----------------
* QUITTER
*----------------

QUITTER JSR  TEXT
        JSR  HOME
        JMP  BASIC

*----------------
* SAISIE NOM FICHIER
*----------------

SAISIN1 JSR  SAISP
        LDY  #>NOMFICH
```

```
        LDA  #<NOMFICH
        BNE  SAISIN3

SAISIN2 JSR  SAISP
        LDY  #>NOMFICH1
        LDA  #<NOMFICH1

SAISIN3 JSR  STROUT
        LDA  #38
        STA  LONG
        LDA  #9
        JSR  GBASCALC
        JSR  INPUT
        BCC  ]1
        RTS
]1      LDY  TPS
        BNE  ]2
        JSR  BIP
        BCS  SAISIN1
]2      STY  PATHNAME
        STY  LNGPATHN
]3      LDA  BUFFIN,Y
        INY
        STA  PATHNAME,Y
        DEY
        DEY
        BPL  ]3
        CLC
        RTS

SAISP   LDA  #7
        STA  CV
        JSR  VTAB
        LDA  #0
        STA  CH
        RTS

NOMFICH ASC
   "Nom du fichier TEXT à créer :"
        DFB  0
NOMFICH1 ASC
   "Nom du fichier TEXT à imprimer :"
        DFB  0

*----------------
* INPUT
*----------------

INPUT   LDX  #0
        STX  POSC

        LDY  LONG
        LDA  #"."
]1      DEY
        BEQ  INCLAV
        STA  (GBASL),Y
        BNE  ]1

INCLAV  LDA  #")"
        STA  (GBASL),Y

INCLAV1 LDA  (GBASL),Y
        EOR  #1
        STA  (GBASL),Y
        LDX  #$39
]11     DEX
        BNE  ]11
        LDA  KBD
        BPL  INCLAV1
        BIT  STROBE

        CMP  #$9B
        BEQ  INESC
        CMP  #$88
        BEQ  INRETOUR
        CMP  #$FF
        BEQ  INRETOUR
        CMP  #$8D
        BEQ  INVALID
        CPY  LONG
        BEQ  INERR
        CMP  #"."
        BCC  INERR
        CMP  #"/"
        BEQ  SLASH
        CMP  #"A"
        BCC  CHIFFRE
        AND  #%11011111
        CMP  #$DB
        BCS  INERR
        BCC  INLETTRE

SLASH   CPY  #0
        BEQ  INLETTRE
        BNE  CHIFFRE1

CHIFFRE CMP  #":"
        BCS  INERR
        CPY  #0
        BEQ  INERR
CHIFFRE1 TAX
        DEY
        LDA  (GBASL),Y
        INY
        CMP  #"/"
        BEQ  INERR
        TXA
        BMI  INLETTRE

INERR   JSR  BIP
        BCS  INCLAV1

INESC   LDY  LONG
        LDA  #" "
]1      STA  (GBASL),Y
        DEY
        BPL  ]1
        LDA  #0
        STA  BUFFIN
        SEC
        RTS

INRETOUR LDA  POSC
        BEQ  INERR
        CMP  LONG
        BNE  INR1
        LDA  #" "
        BNE  INR2
INR1    LDA  #"."
INR2    STA  (GBASL),Y
        DEY
        DEC  POSC
        JMP  INCLAV

INLETTRE STA  (GBASL),Y
        INC  POSC
        INY
        JMP  INCLAV

INVALID STY  TPS
        LDA  #" "
]0      STA  (GBASL),Y
        CPY  LONG
        BEQ  ]1
        INY
        BNE  ]0

]1      LDY  TPS
        LDA  #0
        STA  BUFFIN,Y
        DEY
        BMI  ]3
]2      LDA  (GBASL),Y
        STA  BUFFIN,Y
        DEY
        BPL  ]2
]3      CLC
        RTS

*----------------
* CATALOG
*----------------

CATAL   JSR  HOME
        LDY  #0
]0      LDA  CMD,Y
        STA  $200,Y
        CMP  #$8D
        BEQ  ]1
        INY
        BNE  ]0

]1      JSR  $BE03
        CMP  #6
        BEQ  ]T0
        CMP  #8
        BNE  ]T
]T0     JSR  ERREUR
]T      JSR  GETRET
        JSR  HOME
        JSR  LITPREF
        JMP  CHOIX

*----------------
* CHERCHE DATE/HEURE
* CREE FICHIER
*----------------

CREFIC  LDA  $C061
        ORA  $C062
        STA  POMMES
        LDA  #$C3
        STA  ACCESBT
        LDA  #4
        STA  FILETYPE
        LDA  #0
        STA  AUXFLTP
        STA  AUXFLTP+1
        LDA  #1
        STA  STORAGE

        JSR  CREFICH
        BCC  ]1
        BIT  POMMES
        BMI  ]2
        JSR  ERREUR
        JSR  GETRET
        SEC
]1      RTS
]2      CLC
        RTS

CREFICH JSR  MLI
        DFB  GETTIME
        DA   0
        LDY  #3
]1      LDA  $BF93,Y
        STA  DATECRE,Y
        DEY
        BPL  ]1

        JSR  MLI
        DFB  CREATE
        DA   PARLST
        RTS

POMMES  DS   1

*----------------
* GET EOF
*----------------

GETLONG JSR  MLI
        DFB  GETEOF
        DA   PAREOF
        BCC  ]1
        JSR  ERREUR
]1      RTS

*----------------
* OUVRE FICHIER
*----------------

OUVREFIC JSR  MLI
        DFB  OPEN
        DA   PARLSTO
        BCC  ]1
```

```
        JSR  ERREUR
]1      RTS

*----------------
* ECRIT FICHIER
*----------------

EF      LDA  REFNUM
        STA  REFNUMW
        JSR  MLI
        DFB  WRITE
        DA   PARLSTW
        BCC  ]1
        JSR  ERREUR
]1      RTS

*----------------
* LIT FICHIER
*----------------

LITFIC  LDA  REFNUM
        STA  REFNUMW
        JSR  MLI
        DFB  READ
        DA   PARLSTW
        RTS

*----------------
* VIDE FICHIER
*----------------

VIDEF   LDA  REFNUM
        STA  REFNUM7
        JSR  MLI
        DFB  SETEOF
        DA   PARLST7
        BCC  ]1
        JSR  ERREUR
]1      RTS

*----------------
* CLOT FICHIER
*----------------

CLOSEFIL LDA REFNUM
        STA  REFNUMC
        JSR  MLI
        DFB  CLOSE
        DA   PARLSTC
        RTS

*----------------
* ONLINE
*----------------

ONL     JSR  HOME
        LDY  #>MESONL
        LDA  #<MESONL
        JSR  STROUT
        JSR  MLI
        DFB  ONLINE
        DA   PARLSTN
        BCC  ]0
        JSR  ERREUR
]0      LDA  #15
        STA  CNT
ONLINED LDA  CNT
        ASL
        ASL
        ASL
        ASL
        TAY
        LDA  BUFONLIN,Y
        STA  LNG
        AND  #%00001111
        BEQ  ERR
        LDX  #0
]G      LDA  MESON0,X
        BEQ  ]H
        JSR  COUT
        INX
        BNE  ]G
]H      LDA  LNG
        LSR
        LSR
        LSR
        LSR
        AND  #%00000111
        ORA  #%10110000
        JSR  COUT
        LDX  #0
]E      LDA  MESON1,X
        BEQ  ]F
        JSR  COUT
        INX
        BNE  ]E
]F      LDA  #"1"
        BIT  LNG
        BPL  ]1
        LDA  #"2"
]1      JSR  COUT
        LDA  #" "
        JSR  COUT
        LDA  LNG
        AND  #$F
        STA  LNG
        LDA  #"/"
        JSR  COUT
AFF     INY
        LDA  BUFONLIN,Y
        ORA  #%10000000
        JSR  COUT
        DEC  LNG
        BNE  AFF
        LDA  #$8D
        JSR  COUT
ERR     DEC  CNT
        BPL  ONLINED
        JSR  GETRET
VADEB   JSR  HOME
        JSR  LITPREF
        JMP  CHOIX

*----------------
* GET FILE INFO
*----------------

GFI     JSR  MLI
        DFB  GETFINFO
        DA   PARLSTG
        BCC  ]1
        JSR  ERREUR
]1      RTS

*----------------
* CHANGE PREFIXE
*----------------

MODPREF JSR  HOME
        JSR  LITPREF
MODPR   LDA  #7
        STA  CV
        JSR  VTAB
        LDA  #0
        STA  CH
        LDY  #>MESMOD
        LDA  #<MESMOD
        JSR  STROUT
        LDA  #38
        STA  LONG
        LDA  #9
        JSR  GBASCALC
        INC  GBASL
        JSR  INPUT
        BCS  OKCH
        LDY  TPS
        BNE  MOD0
        JSR  BIP
        BEQ  MODPR
MOD0    STY  PATHNAME
        INC  PATHNAME
        BEQ  MODMLI
        DEC  GBASL
]2      LDA  (GBASL),Y
        INY
        STA  PATHNAME,Y
        DEY
        DEY
        BPL  ]2
MODMLI  JSR  MLI
        DFB  STPREFIX
        DA   PARLSTCP
        BCC  OKCH
        PHA
        JSR  $DAFB
        PLA
        JSR  ERREUR
        JSR  GETRET
        JMP  MODPREF
OKCH    JMP  VADEB
PARLSTCP DFB 1
        DA   PATHNAME

*----------------
* LIT PREFIXE
*----------------

LITPREF JSR  MLI
        DFB  GTPREFIX
        DA   PARLSTCP
        BCC  ]000
        JSR  ERREUR

]000    LDA  #3
        STA  CV
        JSR  VTAB

        LDX  #0
]0      LDA  MESSPR,X
        BEQ  ]00
        JSR  COUT
        INX
        BNE  ]0

]00     LDA  PATHNAME
        STA  LNGPATHN
        LDX  #1
]1      LDA  PATHNAME,X
        ORA  #%10000000
        JSR  COUT
        INX
        DEC  PATHNAME
        BNE  ]1
        JMP  $DAFB

*----------------
* RECEVOIR UN CARAC DANS L'ACC
* CARRY CLEAR SI PAS DE CARACT
*----------------

RECOIT  BIT  DINT
        BMI  RECOIT7

        LDY  REGY
        LDA  $C089,Y
        AND  #%00001000
        BNE  ]1
        CLC
        RTS
]1      LDA  #%00000110
        EOR  $428
        STA  $428
        LDA  $C088,Y
        AND  #%01111111
        SEC
        RTS

RECOIT7 JSR  PARASSC
        LDA  #1
        JSR  STATSSC
        BCS  OK
        RTS

OK      LDA  #%00000110
        EOR  $428
        STA  $428
```

```
        JSR  PARASSC
        JSR  READSSC
        AND  #%01111111
        SEC
        RTS

PARASSC LDX  REGX
        LDY  REGY
        RTS

*----------------
* ERREUR
*----------------

ERREUR  STA  CODERR
        JSR  $DAFB
        JSR  $DAFB
        LDA  #0
        STA  CH
        LDA  #20
        STA  CV
        JSR  VTAB
        LDA  #$21

        JSR  COUT
        LDA  #" "
        JSR  COUT
        JSR  BIP
        LDY  #$FF
]1      INY
        LDA  ERREURS,Y
        CMP  CODERR
        BEQ  ]0
        CMP  #0
        BNE  ]1

]0      INY
        LDA  ERREURS,Y
        BPL  ]0
]2      JSR  COUT
        INY
        LDA  ERREURS,Y
        BMI  ]2
        LDA  #" "
        JSR  COUT
        LDA  #$21
        JSR  COUT
        SEC
        RTS

*----------------
* ATTEND RETURN (OU ESC)
*----------------

GETRET  JSR  $DAFB
        LDA  #21
        STA  CV
        JSR  VTAB
        LDA  #0
        STA  CH
        LDY  #>MESRET
        LDA  #<MESRET
        JSR  STROUT

]1      BIT  KBD
        BPL  ]1

        BIT  STROBE
        LDA  KBD
        CMP  #$D
        BEQ  ]3
        CMP  #$1B
        BEQ  ]4
        JSR  BIP
        BCS  ]1

]3      JSR  ]5
        CLC
        RTS

]4      JSR  ]5
        SEC
        RTS

]5      LDA  #20
        JSR  VTAB
        JSR  CLREOP
        RTS

MESRET  ASC  "
    *** Appuyez sur  <RETURN>  ***"
        DFB  0
```

Source T.POMS.4
Assembleur ProCODE

```
*----------------
* BIP
*----------------

BIP     TYA
        PHA
        TXA
        PHA
        LDY  #$40
BIP1    TYA
        ROR
        TAX
BIP2    DEX
        BNE  BIP2
        BIT  HP
        DEY
        BNE  BIP1
        SEC
        PLA
        TAX
        PLA
        TAY
        RTS

*----------------
* RESET
*----------------

RESET   LDA  #21
        JSR  COUT
        JSR  TEXT
        JSR  HOME
        JSR  CLOSEFIL

        LDY  #0
RESET1  LDA  MESRES1,Y
        BEQ  RESET2
RESET3  JSR  COUT
        INY
        BNE  RESET1

RESET2  JSR  GETRET
        JMP  DEB00

VTAB1   STA  CV
        JMP  VTAB

*----------------
* MESSAGES D'ERREUR
*----------------

ERREURS DFB  3,8,$27,$28,$2F,$5A,$4E
        ASC  "Erreur d'accès disque"
        DFB  6,7,$45,$46
        ASC  "Volume/fichier introuvable"
        DFB  4,$2B
        ASC  "Disque protégé"
        DFB  $40,$44
        ASC  "Nom incorrect"
        DFB  17,$48,$49
        ASC  "Disque/directory saturé"
        DFB  $4E
        ASC  "Protégé en lecture"
        DFB  $7C
        ASC  "Type TEXT seulement..."
        DFB  $7D
        ASC  "Volume saturé"
        DFB  $7F
        ASC  "ProDOS est indispensable"
        DFB  $47
        ASC  "Ce nom de fichier existe déjà"
        DFB  0
        ASC  "Erreur ???"
        DFB  0

*----------------
* INTERFACE //c
*----------------

DEUXC   LDA  #$38
        STA  $47C
        LDA  #$6B
        STA  $47D
        LDA  #$81
        STA  $47E
        LDA  #$7C
        STA  $42
        STA  $3C
        LDA  #4
        STA  $43
        STA  $3D
        STA  $3F
        LDA  #$7E
        STA  $3E
```

Un collaborateur de Pom's vend neuf, pour cause de double emploi, un Moniteur, un stand et une housse – le tout pour Apple //c – au prix tarif Apple —35%.

S'adresser à la revue.

```
         SEC
         JMP  $C311

*----------------
* VERIF SI SSC
*----------------

VERSSC   LDY  #7
VERIF0   TYA
         ORA  #$C0
         STA  VERIF+2
VERIF    LDA  $C00C
         CMP  #$31
         BEQ  ]1
         DEY
         BNE  VERIF0
         SEC
         RTS

]1       STY  INT
         TYA
         PHA
         ORA  #$C0
         STA  SSCINIT+1
         STA  SSCREAD+1
         STA  SSCSTAT+1
         STA  SSC1+1
         STA  ]40+2
         STA  ]2+2
         STA  ]3+2
         STA  ]4+2
         STA  REGX
         PLA
         ASL
         ASL
         ASL
         ASL
         STA  REGY
]2       LDA  $C00D
         STA  SSCINIT
]3       LDA  $C00E
         STA  SSCREAD
]4       LDA  $C010
         STA  SSCSTAT
         LDY  #0
         STY  DINT
]40      LDA  $C000
         CMP  #$2C
         BEQ  ]5
         DEC  DINT
]5       CLC
         RTS

SSC      JMP  (SSC1)
INITSSC  JMP  (SSCINIT)
READSSC  JMP  (SSCREAD)
STATSSC  JMP  (SSCSTAT)

SSCINIT  DA   $0000
SSCREAD  DA   $0000
SSCSTAT  DA   $0000
SSC1     DA   $0000

*----------------
* CHAINES & STOCK
*----------------

MESCHOIX ASC  "RLIMVCQ"

MESMENU  DFB  $8D,$8D
         ASC  "<R>ecevoir annuaire avec n[ de téléphone"
         DFB  $8D
         ASC  "<L>e recevoir sans n[ de téléphone"
         DFB  $8D,$8D
         ASC  "<I>mprimer un fichier"
         DFB  $8D,$8D
         ASC  "<M>odifier le préfixe"
         DFB  $8D,$8D
         ASC  "<V>olumes en ligne"
         dfb  $8D,$8D
         ASC  "<C>atalogue"
         DFB  $8D,$8D
         ASC  "<Q>uitter"
         DFB  $8D
         ASC  "                    Votre choix :"
         DFB  0

MESTITRE ASC  "Annuaire Electroniqu"
         ASC  "e Pom's"
         DFB  $8D,0

POMS     ASC  "  T_Pom's"

PASSSC   ASC  "La carte SuperSérie est nécessaire..."
         DFB  $8D,0

CMD      ASC  "CAT"
         DFB  $8D
PP       DS   1

MESONL   DFB  $8D
         ASC  "Volumes en ligne :"
         DFB  $8D
         ASC  "------------------"
         DFB  $8D,$8D,0
MESON0   ASC  "Port "
         DFB  0
MESON1   ASC  " Lect "
         DFB  0

MESSPR   ASC  "Préfixe "
         DFB  0

MESMOD   ASC  "Nouveau préfixe :"
         DFB  $8D,$8D
         ASC  "/"
         DFB  0

MESRES1  DFB  $8D
         ASC  "     ____________________________"
         DFB  $8D
         ASC  "     "
         INV  "           T_POM'S      V1.0 "
         DFB  $8D,$8D,$8D
         ASC  "     (c) Pom's, CP, JLB  12/5/1987"
         DFB  $8D,$8D,$8D,$8D
         ASC  "     Un programme Pom's,"
         DFB  $8D,$8D
         ASC  "                la revue des Apple"
         DFB  $8D,$8D,$8D
         ASC  "     Revue Pom's : (1) 39 51 24 43"
         DFB  $8D,$8D
         ASC  "     ____________________________"
         DFB  0

DINT     DS   1
POSC     DS   1
LONG     DS   1
TPS      DS   1

PARLST   DFB  7          ; paramètres pour create
         DA   PATHNAME
ACCESBT  DS   1
FILETYPE DS   1
AUXFLTP  DS   2
STORAGE  DS   1
DATECRE  DS   2
         DS   2

PAREOF   DFB  2          ; paramètres pour geteof
         DS   1
         DS   3

INT      DS   1
LNGPATHN DS   1

CODERR   DS   1

PARLSTO  DFB  3          ; paramètres pour open
         DA   PATHNAME
         DA   FILEBUF
REFNUM   DS   1

PARLSTG  DFB  $A         ; paramètres pour getfileinfo
```

```
        DA    PATHNAME
        DS    1          ; lect/écrit/renam...
FILETYG DS    1          ; type de fichier
AUXTYPG DS    2          ; param auxiliaire (lng par ex)
        DS    1          ; type de stockage
BLOCKUS DS    2          ; nb de blocs affectés
        DS    8

PARLSTW DFB   4
REFNUMW DS    1
        DA    $3000
LAS     DS    2
        DS    2

PARLST7 DFB   2
REFNUM7 DS    1
        DFB   0,0,0

PARLSTC DFB   1
REFNUMC DFB   0

PARLSTN DFB   2
        DFB   0
        DA    BUFONLIN

CNT     DS    1
LNG     DS    1

REGX    DS    1
REGY    DS    1
```

```
PARLD0  DFB   2
LD      DS    1
        DA    PATHNAME
PARLD1  DFB   1
        DA    PATHNAME

ACT1    DFB   $23,"#
        DFB   $24,"$
        DFB   $26,"#
        DFB   $27,"]
        DFB   $2C,"<
        DFB   $2D,"^"
        DFB   $2E,">
        DFB   $2F,"v
        DFB   $30,"[
        DFB   $31,"+
        DFB   $38,"/
        DFB   $3C,"/
        DFB   $3D,"/
        DFB   $3E,"/
        DFB   $6A,"*
        DFB   $7A,"*
        DFB   $7B,"B
        DFB   0

ACT2    DFB   $41,$65,"è
        DFB   $42,$65,"é
        DFB   $43,$65,"e
        DFB   $48,$65,"e
        DFB   $41,$61,"à
        DFB   $43,$61,"a
        DFB   $48,$61,"a
        DFB   $43,$69,"i
        DFB   $48,$69,"i
        DFB   $43,$6F,"o
        DFB   $48,$6F,"o
        DFB   $41,$75,"ù
        DFB   $48,$75,"u
        DFB   $4B,$63,"ç
        DFB   0
```

Récapitulation T.POMS

Après avoir saisi cette récapitulation sous moniteur, vous la sauvegarderez par :
BSAVE T.POMS, A$2000, L4033

```
2000:A2 FF 9A AD 00 BF C9 4C
2008:F0 08 A9 7F 20 43 2A 4C
2010:D0 03 20 43 2C 90 16 20
2018:58 FC A0 00 B9 B0 2D F0
2020:09 20 ED FD C8 D0 F5 20
2028:F2 2A 4C D0 03 AD B3 FB
2030:C9 06 D0 21 AD C0 FB D0
2038:03 20 1E 2C AD 3A 2F 20
2040:95 FE A9 07 20 ED FD 20
2048:93 FE A9 07 20 A4 2C 20
2050:3C 2A 20 A7 2C A9 0A 8D
2058:F2 03 A9 2B 8D F3 03 49
2060:A5 8D F4 03 AD 30 BF 8D
2068:6D 2F 20 00 BF C5 6C 2F
2070:B0 35 AD 00 17 29 0F 8D
2078:00 17 A8 B9 00 17 C8 99
2080:00 17 88 88 C0 00 D0 F3
2088:A9 AF 8D 01 17 EE 00 17
2090:20 00 BF C6 70 2F 4C A7
2098:20 4A 21 43 21 10 26 64
20A0:29 C2 28 F3 27 98 26 A9
20A8:15 20 ED FD 20 39 FB 20
20B0:58 FC 20 B5 28 A9 01 20
20B8:36 21 A9 16 20 36 21 A2
20C0:00 BD 8A 2D F0 06 20 ED
20C8:FD E8 D0 F5 20 C7 29 A9
20D0:03 85 22 A9 16 85 23 A9
20D8:04 20 2B 2B A0 08 B9 A7
20E0:2D 99 9F 04 88 10 F7 A9
20E8:04 20 2B 2B A0 00 B9 BF
20F0:2C F0 06 20 ED FD C8 D0
20F8:F5 A9 01 8D 27 2F A9 13
2100:20 47 F8 A9 23 65 26 85
2108:26 90 02 E6 27 20 2C 27
2110:A0 06 AD 00 16 D9 B8 2C
2118:F0 08 88 10 F8 20 F2 2A
2120:B0 D7 A9 99 8D 34 21 98
2128:F0 09 EE 34 21 EE 34 21
2130:88 D0 F7 6C 99 20 20 47
2138:F8 A0 27 A9 DF 91 26 88
2140:10 FB 60 A9 00 8D 77 25
2148:F0 05 A9 01 8D 77 25 20
2150:58 FC 20 C7 29 AD 3B 2F
2158:8D 00 17 A9 04 20 2B 2B
2160:20 58 29 90 06 20 8E 2A
2168:4C A7 20 A9 08 85 24 AD
2170:48 2F 38 ED 4B 2F AA AD
2178:49 2F ED 4C 2F 20 24 ED
2180:E6 24 A2 00 BD 7C 25 F0
2188:06 20 ED FD E8 D0 F5 A9
2190:06 85 22 A0 27 B9 97 25
2198:99 D0 07 88 10 F7 20 A1
21A0:26 90 03 4C A7 20 20 1F
21A8:28 B0 A4 AD 3B 2F 8D 00
21B0:17 20 78 28 90 09 20 43
21B8:2A 20 8E 2A 4C A7 20 20
21C0:A3 28 90 06 20 8E 2A 4C
21C8:A7 20 20 58 FC 20 38 25
21D0:20 96 24 A9 0D 20 2B 2B
21D8:A0 00 B9 BF 25 F0 06 20
21E0:ED FD C8 D0 F5 A9 0A 20
21E8:2B 2B 20 38 25 A9 42 8D
21F0:75 25 A9 FF 8D 74 25 20
21F8:4A 24 C9 1F F0 32 C9 1B
2200:D0 03 4C A6 22 C9 10 D0
2208:03 4C E1 22 C9 18 D0 03
2210:4C 01 23 C9 0C D0 03 4C
2218:36 23 C9 60 D0 03 4C 29
2220:23 C9 16 D0 03 4C 3E 23
2228:AE 76 25 F0 CA 4C 8D 23
2230:AD 71 25 F0 03 20 C2 23
2238:AD 6F 25 F0 03 20 D8 23
2240:AD 6C 25 F0 03 20 E6 23
2248:20 4A 24 8D 69 25 20 4A
2250:24 8D 6A 25 AD 69 25 CD
2258:75 25 B0 03 4C F7 21 D0
2260:1D AD 6A 25 C9 45 D0 13
2268:A9 01 8D 6B 25 A9 00 8D
2270:DB 2D 20 4A 24 20 4A 24
2278:4C F4 23 4C F7 21 AD 6A
2280:25 C9 45 F0 05 B0 17 4C
2288:F7 21 AD 69 25 C9 44 90
2290:0A A9 01 8D 6D 25 A9 00
2298:8D DB 2D 4C F7 21 A9 01
22A0:8D 6C 25 4C F7 21 AD 6D
22A8:25 D0 06 20 4A 24 4C F7
22B0:21 20 4A 24 C9 42 D0 15
22B8:AD 6E 25 F0 0D A9 01 8D
22C0:6F 25 A9 00 8D DB 2D 8D
22C8:6D 25 4C F7 21 C9 47 D0
22D0:0D A9 01 8D 71 25 A9 00
22D8:8D DB 2D 8D 6D 25 4C F7
22E0:21 20 4A 24 C9 02 D0 0A
22E8:A9 01 8D 70 25 A9 00 8D
22F0:DB 2D C9 03 D0 08 A9 00
22F8:8D 70 25 8D DB 2D 4C F7
2300:21 A9 00 8D 6B 25 8D 71
2308:25 8D 6F 25 8D 6D 25 8D
2310:70 25 8D 6C 25 8D 6E 25
2318:8D DB 2D 2C 74 25 30 06
2320:20 33 24 20 72 24 4C F7
2328:21 20 4A 24 C9 12 D0 03
2330:4C 01 23 4C F7 21 A9 00
2338:8D 76 25 4C 01 23 20 4A
2340:24 AA 29 F0 C9 40 F0 18
2348:8A A0 00 D9 73 2F F0 09
2350:C8 C8 C0 22 D0 F5 4C F7
2358:21 C8 B9 73 2F 4C 8D 23
2360:8E 7A 25 20 4A 24 8D 7B
2368:25 A0 00 AD 7A 25 D9 96
2370:2F D0 10 C8 AD 7B 25 D9
2378:96 2F D0 08 C8 B9 96 2F
2380:4C 8D 23 C8 C8 C8 C0 2A
2388:D0 E1 4C F7 21 C9 20 90
2390:2E 8D 72 25 AD 77 25 2D
2398:70 25 8D 70 25 AD 6C 25
23A0:2D DB 2D 0D 71 25 0D 6F
23A8:25 0D 70 25 0D 6D 25 F0
23B0:0E AD 72 25 8D 73 25 29
23B8:7F 8D 74 25 20 2B 24 4C
23C0:F7 21 2C 74 25 30 03 20
23C8:04 25 A9 00 8D 71 25 A9
23D0:01 8D DB 2D 8D 6E 25 60
23D8:2C 74 25 30 03 20 04 25
23E0:A9 00 8D 6F 25 60 2C 74
23E8:25 30 03 20 04 25 A9 00
23F0:8D 6C 25 60 A9 00 8D 41
23F8:25 20 4A 24 C9 1F F0 23
2400:C9 1B F0 0D EE 41 25 AC
2408:41 25 99 41 25 C0 1E D0
2410:E8 A9 00 8D 6B 25 A9 42
2418:8D 75 25 A9 01 8D 76 25
2420:4C F7 21 A9 43 8D 75 25
2428:4C 30 22 09 80 20 DE 24
2430:4C ED FD AD 41 25 D0 01
2438:60 A0 01 B9 41 25 20 2B
2440:24 C8 CC 41 25 D0 F4 4C
2448:04 25 20 FF 29 B0 19 AD
2450:00 C0 2C 10 C0 29 DF C9
2458:D3 F0 0E C9 9B D0 EB 20
2460:1A 25 A2 FF 9A 4C A7 20
2468:60 20 1A 25 A2 FF 9A 4C
2470:EA 21 2C 74 25 30 64 20
2478:04 25 20 42 FC A9 8D 20
2480:DE 24 A0 03 A9 AA 20 DE
2488:24 88 D0 F8 A9 8D 20 DE
2490:24 A0 FF 8C 74 25 A9 08
2498:20 2B 2B A9 00 85 24 A0
24A0:19 A9 AD 20 ED FD 88 10
24A8:F8 A9 AF 20 ED FD A9 A0
24B0:20 ED FD 38 A9 FF E5 18
24B8:AA A9 95 E5 19 20 24 ED
24C0:A9 A0 20 ED FD A9 AF 20
24C8:ED FD A9 AD 20 ED FD A5
24D0:24 D0 F7 20 04 25 A9 0A
24D8:4C 2B 2B 4C 04 25 8D 79
24E0:25 8C 78 25 A5 18 C9 FF
24E8:D0 06 A5 19 C9 95 F0 11
24F0:E6 18 D0 02 E6 19 AD 79
24F8:25 A0 00 91 18 AC 78 25
2500:60 4C 69 24 AD 73 25 C9
2508:8D D0 01 60 A9 8D 8D 73
2510:25 20 DE 24 20 9C FC 4C
2518:FB DA A9 8D 8D 00 30 38
2520:A5 19 E9 30 8D 5A 2F A5
2528:18 8D 59 2F 20 84 28 90
2530:06 20 43 2A 20 8E 2A 60
2538:A9 00 85 18 A9 30 85 19
2540:60 00 00 00 00 00 00 00
2548:00 00 00 00 00 00 00 00
2550:00 00 00 00 00 00 00 00
2558:00 00 00 00 00 00 00 00
2560:00 00 00 00 00 00 00 00
2568:00 00 00 00 00 00 00 00
2570:00 00 00 00 00 00 00 00
2578:00 00 00 00 E2 EC EF E3
2580:F3 A0 EC E9 E2 F2 E5 F3
2588:A0 F3 F5 F2 A0 E3 E5 A0
2590:E4 E9 F3 F1 F5 E5 00 BC
2598:C5 D3 C3 BE A0 D3 E1 F5
25A0:F6 E5 E7 E1 F2 E4 E5 AF
25A8:C6 E9 EE A0 A0 A0 A0 A0
25B0:A0 BC D3 BE A0 D3 E1 F5
25B8:F6 E5 E7 E1 F2 E4 E5 A0
```

```
25C0:A0 A0 A0 A0 A0 A0 A0 A0
25C8:A0 A0 C6 E9 E3 E8 E9 E5
25D0:F2 A0 D4 C5 D8 D4 A0 E3
25D8:F2 FB FB A0 A0 A0 A0 A0
25E0:A0 A0 A0 A0 A0 A0 A0 A0
25E8:A0 D0 F2 E5 F4 A0 F0 EF
25F0:F5 F2 A0 E3 EF EE F3 F5
25F8:EC F4 E1 F4 E9 EF EE A0
2600:E4 E5 A0 EC A7 E1 EE EE
2608:F5 E1 E9 F2 E5 A0 A0 00
2610:20 58 FC 20 C7 29 20 AA
2618:26 90 03 4C A7 20 AD 3B
2620:2F 8D 00 17 20 78 28 90
2628:09 20 43 2A 20 8E 2A 4C
2630:10 26 20 58 29 B0 F2 AD
2638:47 2F C9 04 F0 08 C9 1A
2640:F0 04 A9 7C D0 E3 A9 01
2648:8D 59 2F A9 00 8D 5A 2F
2650:20 96 28 90 0D C9 4C F0
2658:33 20 43 2A 20 8E 2A 4C
2660:A7 20 AD 00 30 09 80 C9
2668:8D D0 05 A0 4F 8C 97 26
2670:20 00 C1 CE 97 26 10 0A
2678:A9 8D 20 00 C1 A0 4F 8C
2680:97 26 AD 00 C0 2C 10 C0
2688:C9 9B D0 C4 A9 8D 20 00
2690:C1 20 93 FE 4C A7 20 00
2698:20 39 FB 20 58 FC 4C D0
26A0:03 20 E1 26 A0 26 A9 ED
26A8:D0 07 20 E1 26 A0 27 A9
26B0:0B 20 3A DB A9 26 8D 27
26B8:2F A9 09 20 47 F8 20 2C
26C0:27 90 01 60 AC 28 2F D0
26C8:05 20 F2 2A B0 D3 8C 00
26D0:17 8C 3B 2F B9 00 16 C8
26D8:99 00 17 88 88 10 F5 18
26E0:60 A9 07 85 25 20 22 FC
26E8:A9 00 85 24 60 CE EF ED
26F0:A0 E4 F5 A0 E6 E9 E3 E8
26F8:E9 E5 F2 A0 D4 C5 D8 D4
2700:A0 C0 A0 E3 F2 FB E5 F2
2708:A0 BA 00 CE EF ED A0 E4
2710:F5 A0 E6 E9 E3 E8 E9 E5
2718:F2 A0 D4 C5 D8 D4 A0 C0
2720:A0 E9 ED F0 F2 E9 ED E5
2728:F2 A0 BA 00 A2 00 8E 26
2730:2F AC 27 2F A9 AE 88 F0
2738:04 91 26 D0 F9 A9 A9 91
2740:26 B1 26 49 01 91 26 A2
2748:39 CA D0 FD AD 00 C0 10
2750:F0 2C 10 C0 C9 9B F0 44
2758:C9 88 F0 51 C9 FF F0 4D
2760:C9 8D F0 6B CC 27 2F F0
2768:2E C9 AE 90 2A C9 AF F0
2770:0C C9 C1 90 0E 29 DF C9
2778:DB B0 1C 90 49 C0 00 F0
2780:45 D0 08 C9 BA B0 10 C0
2788:00 F0 0C AA 88 B1 26 C8
2790:C9 AF F0 03 8A 30 2F 20
2798:F2 2A B0 A5 AC 27 2F A9
27A0:A0 91 26 88 10 FB A9 00
27A8:8D 00 16 38 60 AD 26 2F
27B0:F0 E5 CD 27 2F D0 04 A9
27B8:A0 D0 02 A9 AE 91 26 88
27C0:CE 26 2F 4C 3D 27 91 26
27C8:EE 26 2F C8 4C 3D 27 8C
27D0:28 2F A9 A0 91 26 CC 27
27D8:2F F0 03 C8 D0 F6 AC 28
27E0:2F A9 00 99 00 16 88 30
27E8:08 B1 26 99 00 16 88 10
27F0:F8 18 60 20 58 FC A0 00
27F8:B9 D7 2D 99 00 02 C9 8D
2800:F0 03 C8 D0 F3 20 03 BE
2808:C9 06 F0 04 C9 08 D0 03
2810:20 43 2A 20 8E 2A 20 58
2818:FC 20 C7 29 4C E7 20 AD
2820:61 C0 0D 62 C0 8D 6B 28
2828:A9 C3 8D 2C 2F A9 04 8D
2830:2D 2F A9 00 8D 2E 2F 8D
2838:2F 2F A9 01 8D 30 2F 20
2840:53 28 90 0C 2C 6B 28 30
2848:08 20 43 2A 20 8E 2A 38
2850:60 18 60 20 00 BF 82 00
2858:00 A0 03 B9 93 BF 99 31
2860:2F 88 10 F7 20 00 BF C0
2868:29 2F 60 00 20 00 BF D1
2870:35 2F 90 03 20 43 2A 60
2878:20 00 BF C8 3D 2F 90 03
2880:20 43 2A 60 AD 42 2F 8D
2888:56 2F 20 00 BF CB 55 2F
2890:90 03 20 43 2A 60 AD 42
2898:2F 8D 56 2F 20 00 BF CA
28A0:55 2F 60 AD 42 2F 8D 5E
28A8:2F 20 00 BF D0 5D 2F 90
28B0:03 20 43 2A 60 AD 42 2F
28B8:8D 63 2F 20 00 BF CC 62
28C0:2F 60 20 58 FC A0 2D A9
28C8:DC 20 3A DB 20 00 BF C5
28D0:64 2F 90 03 20 43 2A A9
28D8:0F 8D 68 2F AD 68 2F 0A
28E0:0A 0A 0A A8 B9 00 15 8D
28E8:69 2F 29 0F F0 59 A2 00
28F0:BD 05 2E F0 06 20 ED FD
28F8:E8 D0 F5 AD 69 2F 4A 4A
2900:4A 4A 29 07 09 B0 20 ED
2908:FD A2 00 BD 0B 2E F0 06
2910:20 ED FD E8 D0 F5 A9 B1
2918:2C 69 2F 10 02 A9 B2 20
2920:ED FD A9 A0 20 ED FD AD
2928:69 2F 29 0F 8D 69 2F A9
2930:AF 20 ED FD C8 B9 00 15
2938:09 80 20 ED FD CE 69 2F
2940:D0 F2 A9 8D 20 ED FD CE
2948:68 2F 10 90 20 8E 2A 20
2950:58 FC 20 C7 29 4C E7 20
2958:20 00 BF C4 43 2F 90 03
2960:20 43 2A 60 20 58 FC 20
2968:C7 29 A9 07 85 25 20 22
2970:FC A9 00 85 24 A0 2E A9
2978:1B 20 3A DB A9 26 8D 27
2980:2F A9 09 20 47 F8 E6 26
2988:20 2C 27 B0 34 AC 28 2F
2990:D0 05 20 F2 2A F0 D3 8C
2998:00 17 EE 00 17 F0 0C C6
29A0:26 B1 26 C8 99 00 17 88
29A8:88 10 F6 20 00 BF C6 C4
29B0:29 90 0E 48 20 FB DA 68
29B8:20 43 2A 20 8E 2A 4C 64
29C0:29 4C 4F 29 01 00 17 20
29C8:00 BF C7 C4 29 90 03 20
29D0:43 2A A9 03 85 25 20 22
29D8:FC A2 00 BD 12 2E F0 06
29E0:20 ED FD E8 D0 F5 AD 00
29E8:17 8D 3B 2F A2 01 BD 00
29F0:17 09 80 20 ED FD E8 CE
29F8:00 17 D0 F2 4C FB DA 2C
2A00:25 2F 30 1B AC 6B 2F B9
2A08:89 C0 29 08 D0 02 18 60
2A10:A9 06 4D 28 04 8B 28 04
2A18:B9 88 C0 29 7F 38 60 20
2A20:3C 2A A9 01 20 AD 2C B0
2A28:01 60 A9 06 4D 28 04 8D
2A30:28 04 20 3C 2A 20 AA 2C
2A38:29 7F 38 60 AE 6A 2F AC
2A40:6B 2F 60 8D 3C 2F 20 FB
2A48:DA 20 FB DA A9 00 85 24
2A50:A9 14 85 25 20 22 FC A9
2A58:21 20 ED FD A9 A0 20 ED
2A60:FD 20 F2 2A A0 FF C8 B9
2A68:30 2B CD 3C 2F F0 04 C9
2A70:00 D0 F3 C8 B9 30 2B 10
2A78:FA 20 ED FD C8 B9 30 2B
2A80:30 F7 A9 A0 20 ED FD A9
2A88:21 20 ED FD 38 60 20 FB
2A90:DA A9 15 85 25 20 22 FC
2A98:A9 00 85 24 A0 2A A9 CE
2AA0:20 3A DB 2C 00 C0 10 FB
2AA8:2C 10 C0 AD 00 C0 C9 0D
2AB0:F0 09 C9 1B F0 0A 20 F2
2AB8:2A B0 E8 20 C5 2A 18 60
2AC0:20 C5 2A 38 60 A9 14 20
2AC8:22 FC 20 42 FC 60 A0 A0
2AD0:A0 A0 AA AA AA A0 A0 C1
2AD8:F0 F0 F5 F9 E5 FA A0 F3
2AE0:F5 F2 A0 A0 BC D2 C5 D4
2AE8:D5 D2 CE BE A0 A0 AA AA
2AF0:AA 00 98 48 8A 48 A0 40
2AF8:98 6A AA CA D0 FD 2C 30
2B00:C0 88 D0 F4 38 68 AA 68
2B08:A8 60 A9 15 20 ED FD 20
2B10:39 FB 20 58 FC 20 B5 28
2B18:A0 00 B9 30 2E F0 06 20
2B20:ED FD C8 D0 F5 20 8E 2A
2B28:4C 00 20 85 25 4C 22 FC
2B30:03 08 27 28 2F 5A 4E C5
2B38:F2 F2 E5 F5 F2 A0 E4 A7
2B40:E1 E3 E3 FD F3 A0 E4 E9
2B48:F3 F1 F5 E5 06 07 45 46
2B50:D6 EF EC F5 ED E5 AF E6
2B58:E9 E3 E8 E9 E5 F2 A0 E9
2B60:EE F4 F2 EF F5 F6 E1 E2
2B68:EC E5 04 2B C4 E9 F3 F1
2B70:F5 E5 A0 F0 F2 EF F4 FB
2B78:E7 FB 40 44 CE EF ED A0
2B80:E9 EE E3 EF F2 F2 E5 E3
2B88:F4 11 48 49 C4 E9 F3 F1
2B90:F5 E5 AF E4 E9 F2 E5 E3
2B98:F4 EF F2 F9 A0 F3 E1 F4
2BA0:F5 F2 FB 4E D0 F2 EF F4
2BA8:FB E7 FB A0 E5 EE A0 EC
2BB0:E5 E3 F4 F5 F2 E5 7C D4
2BB8:F9 F0 E5 A0 D4 C5 D8 D4
2BC0:A0 F3 E5 F5 EC E5 ED E5
2BC8:EE F4 AE AE AE 7D D6 EF
2BD0:EC F5 ED E5 A0 F3 E1 F4
2BD8:F5 F2 FB 7F D0 F2 EF C4
2BE0:CF D3 A0 E5 F3 F4 A0 E9
2BE8:EE E4 E9 F3 F0 E5 EE F3
2BF0:E1 E2 EC E5 47 C3 E5 A0
2BF8:EE EF ED A0 E4 E5 A0 E6
2C00:E9 E3 E8 E9 E5 F2 A0 E5
2C08:F8 E9 F3 F4 E5 A0 E4 FB
2C10:EA C0 00 C5 F2 F2 E5 F5
2C18:F2 A0 BF BF BF 00 A9 38
2C20:8D 7C 04 A9 6B 8D 7D 04
2C28:A9 81 8D 7E 04 A9 7C 85
2C30:42 85 3C A9 04 85 43 85
2C38:3D 85 3F A9 7E 85 3E 38
2C40:4C 11 C3 A0 07 98 09 C0
2C48:8D 4D 2C AD 0C C0 C9 31
2C50:F0 05 88 D0 F0 38 60 8C
2C58:3A 2F 98 48 09 C0 8D B1
2C60:2C 8D B3 2C 8D B5 2C 8D
2C68:B7 2C 8D 9A 2C 8D 83 2C
2C70:8D 89 2C 8D 8F 2C 8D 6A
2C78:2F 68 0A 0A 0A 0A 8D 6B
2C80:2F AD 0D C0 8D B0 2C AD
2C88:0E C0 8D B2 2C AD 10 C0
2C90:8D B4 2C A0 00 8C 25 2F
2C98:AD 00 C0 C9 2C F0 03 CE
2CA0:25 2F 18 60 6C B6 2C 6C
2CA8:B0 2C 6C B2 2C 6C B4 2C
2CB0:00 00 00 00 00 00 00 00
2CB8:D2 CC C9 CD D6 C3 D1 8D
2CC0:8D BC D2 BE E5 E3 E5 F6
2CC8:EF E9 F2 A0 E1 EE EE F5
2CD0:E1 E9 F2 E5 A0 E1 F6 E5
2CD8:E3 A0 EE DB A0 E4 E5 A0
2CE0:F4 FB EC FB F0 E8 EF EE
2CE8:E5 8D BC CC BE E5 A0 F2
2CF0:E5 E3 E5 F6 EF E9 F2 A0
2CF8:F3 E1 EE F3 A0 EE DB A0
2D00:E4 E5 A0 F4 FB EC FB F0
2D08:E8 EF EE E5 8D 8D BC C9
2D10:BE ED F0 F2 E9 ED E5 F2
2D18:A0 F5 EE A0 E6 E9 E3 E8
2D20:E9 E5 F2 8D 8D BC CD BE
2D28:EF E4 E9 E6 E9 E5 F2 A0
2D30:EC E5 A0 F0 F2 FB E6 E9
2D38:F8 E5 8D 8D BC D6 BE EF
2D40:EC F5 ED E5 F3 A0 E5 EE
2D48:A0 EC E9 E7 EE E5 8D 8D
2D50:BC C3 BE E1 F4 E1 EC EF
2D58:E7 F5 E5 8D 8D BC D1 BE
2D60:F5 E9 F4 F4 E5 F2 8D A0
2D68:A0 A0 A0 A0 A0 A0 A0 A0
2D70:A0 A0 A0 A0 A0 A0 A0 A0
2D78:A0 A0 A0 A0 D6 EF F4 F2
2D80:E5 A0 E3 E8 EF E9 F8 A0
2D88:BA 00 C1 EE EE F5 E1 E9
2D90:F2 E5 A0 C5 EC E5 E3 F4
2D98:F2 EF EE E9 F1 F5 E5 A0
2DA0:D0 EF ED A7 F3 8D 00 A0
2DA8:A0 D4 DF D0 EF ED A7 F3
2DB0:CC E1 A0 E3 E1 F2 F4 E5
2DB8:A0 D3 F5 F0 E5 F2 D3 FB
2DC0:F2 E9 E5 A0 E5 F3 F4 A0
2DC8:EE FB E3 E5 F3 F3 E1 E9
2DD0:F2 E5 AE AE AE 8D 00 C3
2DD8:C1 D4 8D 00 8D D6 EF EC
2DE0:F5 ED E5 F3 A0 E5 EE A0
2DE8:EC E9 E7 EE E5 A0 BA 8D
2DF0:AD AD AD AD AD AD AD AD
2DF8:AD AD AD AD AD AD AD AD
2E00:AD AD 8D 8D 00 D0 EF F2
2E08:F4 A0 00 A0 CC E5 E3 F4
2E10:A0 00 D0 F2 FB E6 E9 F8
2E18:E5 A0 00 CE EF F5 F6 E5
2E20:E1 F5 A0 F0 F2 FB E6 E9
2E28:F8 E5 A0 BA 8D 8D AF 00
2E30:8D A0 A0 A0 A0 A0 DF DF
2E38:DF DF DF DF DF DF DF DF
2E40:DF DF DF DF DF DF DF DF
2E48:DF DF DF DF DF DF DF DF
2E50:DF DF DF 8D A0 A0 A0 A0
2E58:A0 20 20 20 20 20 20 20
2E60:20 20 20 20 14 1F 10 0F
2E68:0D 27 13 20 20 20 20 20
2E70:20 16 31 2E 30 20 8D 8D
2E78:8D A0 A0 A0 A0 A0 A8 E3
2E80:A9 A0 D0 EF ED A7 F3 AC
2E88:A0 C3 D0 AC A0 CA CC C2
2E90:A0 A0 B1 B2 AF B5 AF B1
2E98:B9 B8 B7 8D 8D 8D 8D A0
2EA0:A0 A0 A0 A0 D5 EE A0 F0
2EA8:F2 EF E7 F2 E1 ED ED E5
2EB0:A0 D0 EF ED A7 F3 AC 8D
2EB8:8D A0 A0 A0 A0 A0 A0 A0
2EC0:A0 A0 A0 A0 A0 A0 A0 A0
2EC8:A0 EC E1 A0 F2 E5 F6 F5
2ED0:E5 A0 E4 E5 F3 A0 C1 F0
2ED8:F0 EC E5 8D 8D 8D A0 A0
2EE0:A0 A0 A0 D2 E5 F6 F5 E5
2EE8:A0 D0 EF ED A7 F3 A0 BA
2EF0:A0 A8 B1 A9 A0 B3 B9 A0
2EF8:B5 B1 A0 B2 B4 A0 B4 B3
2F00:8D 8D A0 A0 A0 A0 A0 DF
2F08:DF DF DF DF DF DF DF DF
2F10:DF DF DF DF DF DF DF DF
2F18:DF DF DF DF DF DF DF DF
2F20:DF DF DF DF 00 00 00 00
2F28:00 07 00 17 00 00 00 00
2F30:00 00 00 00 00 02 00 00
2F38:00 00 00 00 00 03 00 17
2F40:00 10 00 0A 00 17 00 00
2F48:00 00 00 00 00 00 00 00
2F50:00 00 00 00 00 04 00 00
2F58:30 00 00 00 00 02 00 00
2F60:00 00 01 00 02 00 00 15
2F68:00 00 00 00 02 00 00 17
2F70:01 00 17 23 A3 24 A4 26
2F78:A3 27 DD 2C BC 2D A2 2E
2F80:BE 2F F6 30 DB 31 AB 38
2F88:AF 3C AF 3D AF 3E AF 6A
2F90:AA 7A AA 7B C2 00 41 65
2F98:FD 42 65 FB 43 65 E5 48
2FA0:65 E5 41 61 C0 43 61 E1
2FA8:48 61 E1 43 69 E9 48 69
2FB0:E9 43 6F EF 48 6F EF 41
2FB8:75 FC 48 75 F5 4B 63 DC
2FC0:00
```

ProDOS 8 version 1.2

Les versions de ProDOS fleurissent aussi rapidement que les versions de 'Système' sur le Macintosh. ProDOS 8 version 1.2 présente des améliorations appréciables, particulièrement pour les possesseurs d'Apple IIGS. Si votre revue préférée n'est pas encore autorisée à mettre le fichier ProDOS (© oblige...) sur ses disquettes, les revendeurs agréés sont compétents pour le distribuer.

Outre la correction de quelques bugs, la version 1.2 donne l'heure : l'horloge du GS est reconnue, GET_TIME fait maintenant son office normalement. Sur un //e, la carte horloge Thunderclock fonctionnera de 1986 à 1991 grâce à la nouvelle table de ProDOS. La version 1.1.1 se trouvait, elle, limitée à 1987.

Le 'SmartPort' est reconnu comme tel et peut piloter quatre volumes, contre deux avec les versions précédentes ; ils se verront affectés S5/D1, S5/D2, S2/D1 et S2,D2 (ceci n'est vrai que s'il n'y a pas de lecteurs connectés au port 2). Le cas se présente sur le IIGS équipé de deux Unidisks et une carte d'extension mémoire : on trouvera l'Unidisk1 en S5/D1, l'Unidisk2 en S2/D2 et le disque virtuel /RAM5 en S5/D2.

L'ordre de recherche des volumes par ON_LINE a été modifié et, lors d'un démarrage par recherche des ports (option SCAN du tableau de bord), le lecteur 140Ko connecté au port 6 passera après le SmartPort.

Lors du 'boot', ProDOS 8 cherche un éventuel 'driver' du réseau AppleTalk avant de chercher le premier fichier 'système'.

L'octet *MachId* en $BF98 ne permet toujours pas à un programme de lever l'ambiguïté : est-on sur un //e ou un IIGS ? Bien sûr, il est posibble de se décider en testant de nombreux points de la mémoire, à partir de $D742 la routine horloge écrite en 65816 par exemple.

On trouve dans cette version des modifications moins visibles, telles le masquage des interruptions durant les appels au MLI, l'arrivée d'une routine de lecture de blocs simple d'accès...

La volumineuse mise à jour de Beneath Apple ProDOS pour les versions 1.2 et 1.3 donne l'ensemble des modifications avec un détail des adresses et du fonctionnement de ProDOS ; elle est disponible par correspondance auprès de Quality Software ($22,50 franco). VIF à Paris, distribue également les documentations techniques qui font défaut dans les emballages...

La crapule

Enquête policière
Jeu pour Macintosh
512Ko et Plus

Voici encore une création de Jean-Louis Le Breton (responsable et principal auteur de *Froggy Software*) que nous avons beaucoup appréciée, bien que n'étant pas encore parvenus au bout de l'enquête à mener, et après être morts un nombre de fois respectable ! Nous tenons à souligner, outre la qualité du jeu, tant par le scénario que graphisme, trois innovations que l'on aimerait trouver plus fréquemment :

- la disquette ne comporte pas de protection, la documentation expliquant : "Ce jeu a été écrit en Turbo Pascal. Il n'est pas protégé contre la copie. Honte aux pirates. Vive Froggy." ;
- il est possible de s'exprimer en 'petit nègre' (ouvrir porte), ou en bon français (j'ouvre la porte) ;
- une courbe de réussite vous indique à tout moment l'évolution du parcours.

Enfin, vous avez la possibilité de contacter *Froggy Software*, en cas de déprime totale, pour obtenir la solution du jeu. Il y a peu d'éditeurs français ; celui-ci étant en outre de qualité, profitez-en !

Froggy Software
34, avenue Henri Chevreau
75020 Paris

Micro-informations

Jean-Michel Gourévitch

Apple est en passe de gagner son pari. Les Macintosh sont pris au sérieux et s'implantent dans les entreprises. L'annonce des Mac II (qui devraient maintenant commencer à être disponibles au compte-gouttes) a définitivement ouvert les yeux de nombreux chefs d'entreprise qui ne juraient jusqu'ici que par Saint MS-DOS. L'étonnant, c'est que cette consécration du Macintosh se produit précisément au moment où IBM annonce une série de micro-ordinateurs qui représente probablement le standard des cinq prochaines années.

Conséquence : Apple est désormais assuré d'être l'un des survivants de la guerre de la micro-informatique qui n'a pas fini de faire des victimes. Tout est-il donc rose dans le ciel de Cupertino ? Pas forcément. Car il faut bien reconnaître que, malgré son interface imitée par tous et malgré sa souris et ses fenêtres, le Macintosh manque encore d'outils de développement "intelligents" (et notamment des langages type *Prolog* ou *VP-Expert*) qui commencent à se multiplier sur les ordinateurs MS-DOS et permettront de créer les programmes de demain. Il ne faudrait pas que ce retard se prolonge...

Or, voici précisément qu'Apple se dégage quelque peu de l'appui logistique donné aux logiciels. Une filiale a été créée, (au départ à 100%, puis qui deviendra dans un an indépendante) qui reprend la plupart des activités "soft" de la firme à la pomme. Ainsi, Appleworks, MacWrite, MacProject, MacDraw, MacTerminal, etc. ne porteront plus éternellement le logo d'Apple. Officiellement, il s'agit de mettre tous les développeurs sur un même pied d'égalité pour les encourager à mettre au point de nouveaux logiciels. Selon certains, il s'agit aussi de contrebalancer la position dominante de Microsoft, trop liée à IBM. Reste à voir comment se traduira cette nouvelle politique dans les faits. Il est parallèlement intéressant de noter que cette décision avait été anticipée par Guy Kawasaki, "évangéliste du logiciel" chez Apple, et notamment chargé de veiller à ce que tous les logiciels respectent bien l'interface Macintosh. Kawasaki a quitté Apple voici quelques mois pour diriger ACIUS. Derrière cette raison sociale aux consonnances latines se cache tout simplement la filiale américaine d'ACI, qui va vendre aux États-Unis un logiciel dont le nom de code a longtemps été "Silver surfer", mais qui sera finalement lancé sous le nom de "Fourth Dimension". Eh oui, il s'agit bien de *notre* 4D...

Apple se retirerait-il complètement du logiciel ? Pas vraiment. Car, elle gardera dans son verger plusieurs secteurs. D'abord les langages de programmation précédemment évoqués, et pour lesquels il est temps de mettre les bouchées doubles. Ensuite les systèmes. Et notamment Unix. La version A/UX, qui sera proposée sur Macintosh II dès septembre (à condition de rajouter au micro une unité de gestion de la mémoire par page basée sur le coprocesseur Motorola 68851) offrira des possibilités de communication avancée (connexion possible à Ethernet, Appletalk et système de messagerie standard Unix) tout en ayant accès à la boîte à outils du Mac, ce qui permettra donc de développer des applications Unix ayant l'aspect classique des applications Macintosh. Nombreux débouchés possibles dans l'enseignement supérieur et le domaine technique. Et fonctionnement en multitâche...

Le multitâche est encore à l'honneur dans le domaine des systèmes. Car, bien sûr, Apple se garde sous le coude tous les *systèmes*, *finder*, etc. Et voici précisément qu'on travaille à Cupertino sur un nouveau système d'exploitation dont le nom de code est encore "Juggler". Et ce système destiné aux Mac II, serait (à terme, car les premières versions se contenteraient de fonctionner par interruptions) multitâche, utilisant lui aussi le coprocesseur 68851. Se présentant comme le célèbre "switcher", il permettrait de faire tourner simultanément plusieurs programmes, d'effectuer des impressions en tâche de fond, tout en faisant fonctionner un autre logiciel. Bref, un rêve sur micro. Quant au "Servant" développé par Andy Hertzfeld, il ne semble finalement pas qu'Apple l'ait retenu. Hertzfeld pourrait encore le perfectionner et le vendre directement à tous ceux qui n'ont encore que des *vulgaires* Macintosh ou Macintosh Plus (ce qui constitue cependant encore la quasi-totalité des utilisateurs de Mac...).

Chez Apple, on ne cesse de penser à l'avenir. Ce rôle de visionnaire semble particulièrement dévolu à Jean-Louis Gassée. Des propos qu'il a tenu au cours des derniers mois, on peut retenir : l'importance des RNIS (ces nouveaux réseaux de transmission qui permettront de transmettre aussi bien la voix que les données à des vitesses quatre fois supérieures à celles d'aujourd'hui) et des nouveaux moyens de stockages.

La disquette de 800Ko ne tardera pas à devenir aussi obsolète que celle de 143Ko de l'Apple //. Un coup d'œil aux nouveaux *médias* permet de juger la rapidité de l'évolution : les disquettes développées par Bernouilli permettent déjà de stocker 20 Méga-octets, soit l'équivalent d'un volumineux disque dur. Elles sont utilisées par la firme Bering dans son système Totem. Et la firme américaine Jasmine (qui avait déjà présenté un disque dur à 550 dollars, soit moins de 4000,00 F), utilise des disquettes Kodak DTC de 20 Mégas.

Jean-Louis Gassée a également laissé entendre que Bill Atkinson, l'auteur de MacPaint, travaillait à

un produit révolutionnaire utilisant une interface encore plus conviviale pour remplacer le clavier. Avec la mise sur le marché du IIGS et du Macintosh II, la nécessité d'une impression en couleurs est devenue plus aiguë. On travaille donc chez Apple d'arrache-pied à la mise au point d'une véritable imprimante couleurs (utilisant la technologie du jet d'encre ou celle du laser) à un prix restant abordable. Pas si facile.

Enfin, il va falloir assurer l'avenir du Macintosh II. Jean-Louis Gassée a annoncée qu'Apple allait publier les spécifications du Nubus, permettant ainsi aux fabricants de cartes et aux développeurs de construire des cartes intelligentes contenant des ROM communiquant directement entre elles sans recours au processeur. Le Mac II a d'ailleurs déclenché une frénésie d'annonces de programmes et de cartes d'extension.

Le Macintosh II au programme

BrainPower, l'auteur de Statview – l'un des meilleurs programmes de statistiques pour le Mac – a annoncé une nouvelle version pour le II utilisant le coprocesseur 6881, l'accélérateur de virgulle flottante, et capable d'adresser 2 Gigaoctets de mémoire. Ce programme permettant des régressions à variables multiples, etc. sera vendu dès juillet pour 450 dollars.

Côté cartes d'extensions, c'est la surnatalité. **Supermac Technology**, propose des cartes graphiques (1365 x 1024 points en monochrome et 1024 x 768 en couleurs). Tout comme **AST Research**, qui a développé des cartes d'extension mémoire, des cartes sorties, et la carte 386 permettant une émulation MS-DOS. **Sigma Designs**, a développé **Laserview**, un moniteur monochrome et son adaptateur permettant une résolution de (1664 x 1200 points) vendus 2300 dollars. Cette carte fonctionne également sur le Mac SE. Et ce n'est qu'un début. Au programme, pour très bientôt : d'autres cartes avec co-processeurs pour faire fonctionner des programmes MS-DOS, un déluge de cartes vidéo, communication ou mémoire, des systèmes d'acquisition des données, ou de contrôles de processus pour environnements scientifiques ou de production en usines, des cartes musicales, etc.

Macintosh Plus, toujours du graphisme

Sur le front du Macintosh Plus, la source des nouveautés ne se tarit pas encore. Voici d'abord, importé chez nous par **P-Ingéniérie**, l'**Adobe Illustrator** d'Adobe (créateur du langage PostScript). Un véritable outil de création graphique, bien sûr basé sur le langage de description graphique PostScript, qui définit les images à l'aide de droites et de courbes précises et non de points. On peut utiliser comme base de travail une photographie numérisée, un croquis ou un document MacPaint, et ensuite la transformer, la reprendre, l'enrichir, ajouter des légendes, pivoter, agrandir, réfléchir, déformer certains détails. Le résultat final peut être imprimé sur une imprimante Laserwriter, ou une Linotronic 300 (définition de 2540 points par pouce). Pour les tirages couleurs, Adobe Illustrator imprime séparément les quatre films couleurs. À côté de ce véritable programme graphique professionnel, MacPaint a soudain l'air d'un jouet pour élèves de maternelle. Si ce logiciel vous intéresse, Pom's peut vous faire parvenir contre 80,00 F, une disquette de démonstration en anglais qui en dit long sur les possibilités de ce programme vendu 5400 F HT.

Laserware, a d'autre part développé aux États-Unis un autre programme graphique générant automatiquement du code PostScript et produisant des films séparés pour les couleurs : c'est **Laserpaint**, composé d'un module de dessin, d'un module de peinture, et d'un modèle de texte permettant le crénage, etc. Prix 495 dollars.

Toujours de la PAO...

Sur le front de la publication assistée par ordinateur, on ne désarme pas. Tandis que PageMaker sort sa version 2.0, voici qu'apparaît **Scoop**, de **Target Software**, un logiciel de mise en page (jadis connu sous le nom de Solo), qui promet de ne pas limiter ses utilisateurs (comme ceux de Page Maker) à des formats rigides et à des images rectangulaires, le texte s'enroulant automatiquement autour d'images même irrégulières. On peut aussi créer simultantément les textes et les graphiques.

Voici encore **Xpress** de **Quark**, riche de nombreuses possibilités typographiques, de crénage de précision, au 1/100e de la largeur d'un caractère, d'habillage d'illustrations irrégulières, etc. Bref, un logiciel dont on reparlera à coup sûr.

À remarquer, encore la version III de **MacPublisher**, de **Boston Publishing Systems**, que Lettraset n'a finalement pas adoptée, lui préférant **Ready Set Go 3.O**. Cette nouvelle version considérablement améliorée, et fonctionnant en couleurs sur le Macintosh II possède des niveaux de gris réglables par incréments de 10%, des possibilités de rotations, une césure automatique, etc.

Enfin, **Office Talk** est à la fois un système de communication et de publication, compatible MS-DOS et Macintosh, qui permet de s'échanger des fichiers, de les mettre en pages, d'expédier du courrier électronique et d'imprimer les documents sur Linotype. Un système dont nous reparlerons en détails.

Amélioration de textes

Voici l'**Orthogiciel Plus**, la dernière mouture du correcteur orthographique de **Larousse**. À remarquer que, contrairement aux versions précédentes, celle-ci fonctionne avec MacWrite 4.5, ou n'importe quel fichier texte. On peut se constituer des dictionnaires particuliers (c'est même fortement conseillé car, en examinant un texte, le programme s'arrête sur tous les noms propres ou termes qu'il ne connaît pas). Le fonctionnement est impressionnant et impeccable. Un outil indispensable à tous ceux qui veulent éviter les fautes d'orthographe. Un seul regret : que ce programme ne fonctionne pas au moyen d'un accessoire de bureau, ce qui permettrait de lui faire corriger un texte sans avoir à quitter le traitement de texte utilisé. Enfin, il reste toujours la possibilité de le faire fonctionner sous Switcher...

Bootware Software Company, a créé Personal **Résumé-Writer**, qui aide à écrire... non pas des résumés (gare au contre-sens), mais des curriculum vitæ. On tape les informations dans des fenêtres pré-définies, et le programme met le document en page conformément aux spécifications qui ont été précisées. Prix : 30 dollars pour un seul nom et 595 dollars pour un nombre illimité de noms (pour les professionnels du CV ou du recrutement).

Powerpoint de **Forethought Inc.** permet, lui, comme More, de réaliser des diagrammes pour présentations d'organigrammes, cartes, etc. Les graphiques et textes peuvent être importés d'autres programmes Macintosh, les écrans sont imprimés sur Laserwriter. Prix : 395 dollars.

Logiciels spécialisés

Pour les actions, **The Right time** de **TBSP** est un logiciel graphique d'aide à la décision boursière donnant des indications sur le moment d'achat ou de vente. Prix : 399 dollars. Est-ce le bon moment pour l'acheter ?

Ciné Mac

Beck-Tech a créé **Mac-Movies**, un programme spectaculaire, permettant d'animer des images, plus simple que VideoWorks et vendu 99 dollars. Mac Movies est composé de quatre modules, permettant de visualiser une série de documents MacPaint, de comprimer les documents et d'enchaîner les documents ensemble. On peut les voir sous forme de films et relier des films les uns aux autres. À noter que le Mac commence à envahir les studios d'Hollywood. On trouve ainsi des logiciels permettant comme **ScriptWriter** d'écrire des scripts sur deux colonnes (avec numération automatique des scènes, placement des dialogues, etc) et **Story Boarder**, un programme de dessin permettant de programmer les séquences d'un film avec les effets de transitions, etc.

Des utilitaires

Ce sont les programmes accessoires qui rendent la vie plus facile. Les utilisateurs de Mac les connaissent bien. Voici d'abord des *templates*, ces formes prédéfinies qu'il ne reste plus qu'à remplir, ou ces programmes déjà mâchés. Dans cette dernière catégorie, **TBI** a réalisé avec l'aide de 4D un programme de gestion agricole. Dans la première, **Aldus**, l'éditeur de PageMaker, distribué en France par ISE Cegos, a étudié un **Portfolio**. Il s'agit de maquettes prédéfinies pour des mises en page avec PageMaker. Il ne reste plus qu'à insérer son texte ou ses images dans une maquette qui a été dessinée par des "pros". Bien commode pour tous ceux qui doivent faire de l'édition électronique mais ne connaissent pas forcément l'art de la mise en page.

L'impression est aussi un domaine de choix pour les utilitaires. **Softstyle** en est devenu un spécialiste. Il propose ainsi **Laserstart Plus**, permettant pour 145 dollars d'utiliser avec le Mac une imprimante HP Laserjet, Canon LBP 8A, etc. Quant à **Colormate**, c'est un logiciel qui permet d'imprimer en couleurs des documents du Mac sur l'imprimante ImageWriter II.

Un autre système plus onéreux, **Ps Jet** de **Laser Connection** permet pour 3000 dollars de transformer une imprimante Laser HP Laserjet en Laserwriter. Seul problème : l'ensemble revient alors plus cher que l'imprimante d'Apple...

Quant à **KroyKolor** de **Kroy Sign Systems**, c'est un appareil permettant de colorier un document. On l'imprime, puis on le glisse dans une feuille de film transfert, on le place dans un appareil et le document se colorie. On peut aussi réaliser des documents en plusieurs couleurs,

Universal File Conversion

De nombreux lecteurs cherchent vainement ce programme chez les revendeurs français. Il est possible de le commander directement aux États-Unis ($44,95 franco, indiquer le numéro de carte VISA) auprès de :
Quality SoftWare, 21601 Marilla Street, Chatsworth, California 91311.
Rappelons que ce programme permet de convertir les fichiers d'un système d'exploitation à l'autre : DOS 3.3, ProDOS/SOS, CP/M et Pascal.

en découpant des morceaux de films transfert. La machine coûte 995 dollars et les films 50 cents la page.
À noter enfin des programmes qui peuvent rendre bien des services : la version 2 de **My Disk Labeler** de **Williams & Macias** imprime des étiquettes pour les disquettes éventuellement en couleurs ou sur Laserwriter en mêlant le graphique au texte. Prix : 45 dollars.
Et **DiskEnsure** et **Disk-Express** de **ALSoft** permettent de réaliser des copies de sauvegarde d'un disque dur, et de réorganiser les fichiers pour accélérer les accès. Pour 40 dollars chacun.

Côté matériel

Remarquable le disque dur **DataFrame 40XP** de **SuperMac**. C'est tout simplement le plus rapide, à ce jour, des disques dur pour le Mac. Un 40 mégas pour 1900 dollars. Puissant le **Turbo Max** de **Mac Memory**, un accélérateur pour le MacPlus qui le fait fonctionner trois fois plus vite avec un coprocesseur 68881 et 1,5 Megas de mémoire pour 1300 dollars.
Versatile, le **coprocesseur MS-DOS** pour MacPlus de **Dayna Mac**, le créateur du célèbre MacCharlie, Pour 795 dollars, on a un lecteur de disquettes de 360Ko, une carte avec 256Ko de RAM extensible, une copie de MS-DOS. La carte se branche dans un des ports série et permet de faire fonctionner les logiciels MS-DOS dans une fenêtre. Ingénieux, l'accélérateur de **Peak Systems**, permettant au Mac SE, grâce à une carte d'utiliser le processeur 68020 du Mac II. Même chose pour le **Prodigy SE** de **Levco. Sensible**, le **Data Acquisition** de **GW Instruments**, un boîtier qui permet de nourrir le Macintosh de données recueillies à l'extérieur, et de le transformer en oscilloscope, analyseur de spectre, etc.

Et l'Apple // alors ?

Revoici des programmes pour ce bon vieil Apple //. Sa version GS lui a décidément donné un coup de jeune. **Word Perfect** lui a ainsi transposé son traitement de textes. **Bank Street Writer** de **Scolastic Software**, qui fut un des premiers traitements de textes faciles en est à sa version III et comprend un dictionnaire de correction (en anglais seulement hélas). Prix : 100 dollars.

Du graphisme

Paintworks Plus d'**Activision** est un programme de dessin doté d'une fonction d'animation pour réaliser des dessins animés. La brosse du pinceau dispose ainsi de quelque 32 formes, on peut utiliser 16 couleurs sur une seule image, choisies dans une palette de 4096, etc. Prix : 80 dollars. **TopDraw** de **Styleware** est un mélange de MacDraw pour le Mac et de MacPaint, avec les mêmes possibilités de couleurs.

Du son

Music Studio d'**Activision** permet de bénéficier des capacités sonores du GS. On peut composer des partitions, utiliser 45 instruments prédéfinis ou en créer de nouveaux, le tout sur 15 canaux. Prix : 80 dollars.

Des sciences

Le **Science Toolkit** de **Broderbund Software** correspond à la boîte de chimie qu'on offrait jadis aux enfants. Différence, tout se passe sur l'écran, mais avec des instruments de mesure qui fournissent, par exemple, la température à l'ordinateur. On doit disposer d'un module maître. Ensuite il y a deux modules, l'un consacré à la vitesse et à l'action, le second à l'activité séismique. Prix de chacun des deux modules : 40 dollars.

Du matériel

Il n'y avait pas de raison pour que l'Apple // ne bénéficie pas d'un numériseur d'images comme le Mac. À l'image du Thunderscan, le **Scannit** de **JED Design** s'installe à la place du ruban de l'imprimante Imagewriter et permet de numériser une image. Prix : 219 dollars. Pas de raisons non plus pour que l'Apple // ne puisse pas lire les fichiers MS-Dos. C'est chose faite avec la carte **the Envoy** d'**Asky**. Ce contrôleur permet d'écrire et de lire des disquettes au format MS-DOS. À condition, bien sûr, de lui relier un lecteur de disquettes compatible. Prix du contrôleur : 180 dollars. La firme travaille à des utilitaires permettant de convertir les données de Lotus en ProDOS. Pas de raisons non plus qu'on ne puise utiliser l'Apple // en usine. On le sait notamment en France chez Peugeot. **Isaac**, de **Cyborg**, est un boîtier d'acquisitions de données permettant de contrôler des processus, de transformer l'Apple en spectroscope, chromatographe, etc.

Et il y avait encore moins de raisons de ne pas transformer l'Apple IIGS en un instrument stéréo haute fidélité. C'est fait avec la carte **Supersonic** et **Supersonic Digitizer** de **MDIdeas**. On branche cette carte sur des enceintes acoustiques, et on obtient un son stéréo. Prix : 60 dollars et 40 dollars pour le numériseur. La même firme propose une carte d'extension de mémoire jusqu'à 8 Mégas permettant d'utiliser les programmes stockés en RAM et préservés par une pile rechargeable. Prix : 350 dollars pour la carte d'un méga et 180 dollars pour la carte **ESP rampack**. Dans le même esprit, voici la carte **Speedisk** de **Thot Informatique** : compatible][+, //e, IIGS, il s'agit d'un disque virtuel à mémoire permanente sauvegardée jusqu'à deux mois grâce à sa pile. De 384Ko à 1 Méga, les transferts se font à

50Ko à la seconde sur le GS pour un prix allant de 4 à 6000,00 F selon capacité.

Adresses

Bering Industries 280 Technology Circle Scotts Valley CA 95066

Jasmine Computer Systems 555 De Haro St, San Francisco CA 94107

BrainPower, 24009 Ventrura Blvd, Suite 250, Calabasas, CA 91302

Supermac Technology, 950 N Rengstorff Ave. Mountain View, CA 94043

AST Research, 2121 Alton Ave Irvine, CA 92714

Sigma Designs, 46501 Landing Parkway, Fremont CA 94538

Dayna Communications, 50 S. Main St, Salt Lake City UT 84144

P-Ingéniérie, 226 Bd Raspail, 75014 Paris, Tel. : 43 21 93 36

Laserware, PO Box 668, San Rafael, CA 94915

Target Software, 14206 SW 136th St, Miami, FL 33186

Boston Publishing Systems, 1260 Boylston St Boston MA 02215

Bootware Software Company, 5856 Parkmor Rd Calabasas CA 91302

Forethought Inc. 250 Sobrante Way, Sunnyvale, CA 94086

TBSP, 8821 Alcott St, Los Angeles, CA 90035

Beck Tech, PO Box 5027 Berkeley CA 94705

Softstyle 7192 Kalanianaole Hwy Suite 205 Honolulu, Hawaï 96825

Laser Connection, PO Box 850296 Mobile AL 36685

Kroy Sign Systems 7560 East Redfield Road Scottsdale Arizona 85260

Williams & Macias PO Box 19206 Spokane WA 99219

ALSoft PO Box 927 Spring Tx 77383

Mac Memory 2480 North First Street, San Jose, CA 95131

Peak Systems 1201 Spyglass Austin TX 78746

Levco 6161 Lusk Blvd Suite C 203 San Diego CA 92121

GW Instruments, PO Box 2145 Cambridge MA 02141

Activision PO Box 7287 Mountain VIew CA 94039

Styleware 5250 Gulfton Suite 2E Houston Texas 77081

Broderbund Software 17 Paul Drive San Rafael CA 94903

JED Design 3300 Central Ave SE Canton OH 44707

Asky, 4320 Stevens Creek Blvd Suite 287 San Jose CA 95129

Cyborg 55 Chapel St Newton, MA 02158

MDIdeas, 1163 Triton Drive Foster City CA 94404

Thot Informatique B.P. 421 49004 Angers Cedex Tél : 41 86 17 74

Polices de la disquette Pom's Mac B

Portez ce bol de vieux whisky aux 257 juges qui fument

PORTEZ CE BOL DE VIEUX WHISKY AUX 257 JUGES QUI FUMENT

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky

Portez ce bol de vieux whisky aux

Portez ce bol de vieux whisky aux 257

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui

Portez ce bol de vieux

Portez ce bol de vieux whisky aux

Portez ce bol de vieux whisky aux 257 juges

Polices de la disquette Pom's Mac C

PORTEZ CE BOL DE VIEUX WHISKY AUX 257 JUGES

Portez ce bol de vieux whisky aux

Portez ce bol de vieux

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 25

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 25

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges

Portez ce bol de vieux whisky aux 257

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

PORTEZ CE BOL DE VIEUX WHISKY AUX 257

Les disquettes Mac B à Mac I

Ces disquettes contiennent des programmes ou documents 'domaine public' ou 'Freeware' : elles sont le résultat d'une sélection coinjointe de Pom's et du Club Apple parmi les 25 disquettes initialement proposées par le Club Apple. Cette réduction de 25 disquettes en 8 correspond à une élimination de nombreux programmes qui soit ne fonctionnaient pas sur Macintosh Plus, soit présentaient un intérêt pratique réduit. Vous trouverez ci-dessous un 'catalogue' des disquettes avec, pour chaque article, une brève description.

☞ Disquette Mac B – Polices de caractères

Andover 12 – APL 12 – ASCII 12 – Bookman* 14/24 – Boxes 9/10/12 – Broadway 24 – Century 18/24 Chancery 24 – Chicago by night 12 – Circus 12/18/24/36 – Cirth 48 – Cursive 12 – Cyril 12 East Orange 12 – Elite 12 – Elvish 12 – Eon 12 – Exeter 13 – Greek 9/10/12/14/18/24 Hollywood 12/24 – Hood River 12 – Lineal 18 – Mars 18 – London 18/36 – Long Island 12/18.

☞ Disquette Mac C – Polices de caractères

N Helvetica Narrow* 14 – Ophir 12/24 – Palo Alto 9/12/18/24/36 – Park Avenue 18 Philly 9/10/12/18/20/24 – Pica 12 – Princeton 12/24 – Ravenna 12/24 – Runes 12/24 – Santiago 12 Silicon Valley 12 – Stencil 12/24 – Symbol* 9/12/18/24 – Tiny 12 – Toronto fixe 9 – Vectors 9/12 Zapf Dingbats* 18.

** les polices marquées d'une étoile ne sont pas celles de la 'LaserWriter' portant les mêmes noms.*

☞ Disquette Mac D – Accessoires de bureau

ArtThief pour visualiser des images MacPaint ou FullPaint, copier des images plus grandes que les fenêtres de ces applications... tout ceci sans quitter l'application en cours.

Ascii.DA pour connaître immédiatement le code ASCII d'un caractère, en décimal, en hexadécimal et en octal.

FKEY à la fois accessoire et application. Il permet l'installation de 'Mini-programmes' semblables, dans l'esprit, aux accessoires de bureau, mais accessibles depuis une touche de fonction tout comme 'Commande-1' provoque l'éjection de la disquette du lecteur interne, ou 'Commande-3' la copie de l'écran sur disquette.Voici la liste des fonctions fournies sur la disquette Mac D :

- *ASCins* pour saisir n'importe quel caractère, même s'il n'est pas accessible directement au clavier.
- *ASClip* idem, mais le caractère est placé dans le 'presse-papiers'.
- *BigCursor* la taille des curseurs est doublée.
- *Clippy* place dans le 'presse-papiers' les caractères spéciaux.
- *ComKey* pour lire la zone 'commentaire' des fichiers. Ne fonctionne pas en HFS.
- *DA Key* pour essayer un accessoire sans l'installer dans le menu .
- *FadeKey* pour 'éteindre' momentanément l'écran.
- *PanicKey* le Mac se défoule à votre place !
- *ScreenToClip* place la partie de l'écran de votre choix dans le 'presse-papiers'.
- *SetSound* pour régler le niveau sonore.
- *ShowKey* 'catalogue' des touches utilisées et de leurs fonctions.
- *Sleep* proche de FadeKey mais avec une .

Poor Boy's création et installation de macro-instructions.

Dec Maker pour les décideurs en panne d'inspiration ; cet accessoire ludique prend les décisions à leur place !

File Tools permet diverses manipulations de fichiers.

Maxwell un jeu d'adresse et de patience.

TheBox à la fois montre, chronomètre, et indicateur de mémoire libre.

☞ Disquette Mac E – Applications

AMAZING	jeu de labyrinthe, avec plusieurs niveaux de difficulté.
Analog Clock	horloge 'à aiguilles'.
Backgammon	célèbre jeu de société, très bien présenté.
Banner	impression de banderoles sur papier 'listing'.
Bricks	jeu 'casse briques'.
Canfiels	jeu de carte.
DiskUtil	utilitaire pour disquettes. Contrôle, etc.
Fast Formatter	Formatage rapide de disquettes
MouliMac	Conversion de fichiers au format SYLK (Excel, Multiplan, 4ème Dimension) en format 'texte'.

☞ Disquette Mac F – Applications

DataFlow	programme sophistiqué pour le dessin d'algorythmes.
Reversi	jeu de reflexion.
yapu	utilitaire pour ImageWriter.
Enigma	superbe jeu du type Rubik cube, mais avec deux cercles croisés.
F/DA Sorter	tri alphabétique des articles dans les menus.
File Diddler	manipulation de fichiers 'texte'. Ajoute ou enlève certains caractères de contrôle.
HexPuzzle	jeu sur une valeur hexadécimale.
Iago	jeu de reflexion.

☞ Disquette Mac G – Applications

ResEdit	éditeur de ressources.
Icon Editor	éditeur d'icônes et curseurs.
JClock	installe l'heure dans la barre de menu. Elle y reste même si vous changez d'application.
Life	jeu de la vie extrêmement rapide écrit par l'auteur de MacPaint.
Missile Command	jeu d'action.

☞ Disquette Mac H – Applications

MacBILLBOARD	éditeur d'affiches. Permet l'agrandissement et l'impression d'images MacPaint jusqu'à 32 fois.
MacWait	remplace le curseur en forme de montre par une montre animée.
MenuEdit	éditeur de menus.
PackIt	compacteur de fichiers
Social Climber	jeu d'adresse.
Solitaire	jeu de reflexion.
WaveMaker	éditeur de sons évolués.

☞ Disquette Mac I – Applications

Snooker	jeu de billard très réaliste et comportant de nombreuses options : billard français, américain, nombre de billes, angle d'attaque variable, etc.
SystemVersion	donne le numéro de version et la date de création du fichier système courant.
Adventure of Snake	jeu d'adresse.
ThinkTank to WP	conversion de fichiers ThinkTank en fichiers 'texte' utilisable depuis un logiciel de traitement de textes.

Disquettes Apple][

E.P.E. 5.1 - 140Ko	200,00 F	☐
E.P.E. 5.1 - 800Ko	200,00 F	☐
Échange E.P.E. 5.0/5.1 - 140Ko	80,00 F	☐
Échange E.P.E. 5.0/5.1 - 800Ko	80,00 F	☐
Pascal - 140Ko	80,00 F	☐
Max (moniteur étendu) - 140Ko	150,00 F	☐
Dominos - 140Ko	80,00 F	☐
COGO - 140Ko	200,00 F	☐
Ludologic - 140Ko	80,00 F	☐
Ordico - 140Ko	200,00 F	☐
BananaSoft - 140Ko	200,00 F	☐

Recueils Pom's

Numéro 1 (Revues 1 à 4)	140,00 F	☐
Disquettes **A][** 140Ko **1 à 4**	200,00 F	☐
Numéro 2 (Revues 5 à 8)	140,00 F	☐
Disquettes **A][** 140Ko **5 à 8**	200,00 F	☐
Numéro 3 (Revues 9 à 12)	140,00 F	☐
Disquettes **A][** 140Ko **9 à 12**	200,00 F	☐
_____ **reliures toilées** pour 6 numéros, soit un an	60,00 F	☐

Apple][800Ko 3'5 numéro **29**	80,00 F	☐
Apple][800Ko 3'5 numéro **30**	80,00 F	☐

Macintosh

Mac Raccourci	200,00 F	☐
MacAstuces	200,00 F	☐
Mac 'A'	80,00 F	☐
Mac 'B'	80,00 F	☐
Mac 'C'	80,00 F	☐
Mac 'D'	80,00 F	☐
Mac 'E'	80,00 F	☐
Mac 'F'	80,00 F	☐
Mac 'G'	80,00 F	☐
Mac 'H'	80,00 F	☐
Mac 'I'	80,00 F	☐
Mac 14/15/16	150,00 F	☐
Mac 17	80,00 F	☐
Mac 18	80,00 F	☐
Mac 19	80,00 F	☐
Mac 20	80,00 F	☐
Mac 21	80,00 F	☐
Mac 22	80,00 F	☐
Mac 23	80,00 F	☐
Mac 24	80,00 F	☐
Mac 25	80,00 F	☐
Mac 26	80,00 F	☐
Mac 27	80,00 F	☐
Mac 28	80,00 F	☐
Mac 29	80,00 F	☐
Mac 30	80,00 F	☐

Accompagnement

A][140Ko **1/2**	60,00 F	☐
A][140Ko **3**	60,00 F	☐
A][140Ko **4**	60,00 F	☐
A][140Ko **5**	60,00 F	☐
A][140Ko **6**	60,00 F	☐
A][140Ko **7**	60,00 F	☐
A][140Ko **8**	60,00 F	☐
A][140Ko **9**	60,00 F	☐
A][140Ko **10**	60,00 F	☐
A][140Ko **11**	60,00 F	☐
A][140Ko **12**	60,00 F	☐
A][140Ko **13**	60,00 F	☐
A][140Ko **14**	60,00 F	☐
A][140Ko **15**	60,00 F	☐
A][140Ko **16**	60,00 F	☐
A][140Ko **17**	60,00 F	☐
A][140Ko **18**	60,00 F	☐
A][140Ko **19**	60,00 F	☐
A][140Ko **20**	60,00 F	☐
A][140Ko **21**	60,00 F	☐
A][140Ko **22**	60,00 F	☐
A][140Ko **23**	60,00 F	☐
A][140Ko **24**	60,00 F	☐
A][140Ko **25**	60,00 F	☐
A][140Ko **26**	60,00 F	☐
A][140Ko **27**	60,00 F	☐
A][140Ko **28**	60,00 F	☐
A][140Ko **29**	60,00 F	☐
A][140Ko **30**	60,00 F	☐

Revue n° 4	35,00 F	☐	**Revue n° 7**	35,00 F	☐	**Revue n° 8**	35,00 F	☐
Revue n° 10	40,00 F	☐	**Revue n° 11**	40,00 F	☐	**Revue n° 12**	40,00 F	☐
Revue n° 13	40,00 F	☐	**Revue n° 14**	40,00 F	☐	**Revue n° 15**	40,00 F	☐
Revue n° 16	40,00 F	☐	**Revue n° 17**	40,00 F	☐	**Revue n° 18**	40,00 F	☐
Revue n° 19	40,00 F	☐	**Revue n° 20**	40,00 F	☐	**Revue n° 21**	40,00 F	☐
Revue n° 22	40,00 F	☐	**Revue n° 23**	40,00 F	☐	**Revue n° 24**	40,00 F	☐
Revue n° 25	40,00 F	☐	**Revue n° 26**	40,00 F	☐	**Revue n° 27**	45,00 F	☐
Revue n° 28	45,00 F	☐	**Revue n° 29**	45,00 F	☐	**Revue n° 30**	45,00 F	☐

Abonnements pour six numéros à partir du _____, à :

la **revue** seule	225,00 F	☐
la **revue** et les **disquettes Apple][** 140Ko – 5' 1/4	525,00 F	☐
la **revue** et les **disquettes Apple][** 800Ko – 3' 1/2	625,00 F	☐
la **revue** et les **disquettes Macintosh**	625,00 F	☐
la **revue**, les **disquettes Apple][** 140Ko – 5' 1/4 et les **disquettes Macintosh**	925,00 F	☐
la **revue**, les **disquettes Apple][** 800Ko – 3' 1/2 et les **disquettes Macintosh**	1025,00 F	☐

* pour les envois par avion, ajoutez 15 F par numéro et/ou par disquette soit, par exemple, 90 F pour un abonnement avec disquettes.

Envoyez ce bon et votre règlement à : **Éditions MEV – 12, rue d'Anjou – 78000 Versailles**

Nom : _____

Adresse : _____

Règlement par : Carte Bleue/VISA ☐ Chèque bancaire ☐ Chèque postal ☐ Mandat ☐

numéro de la carte _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ date d'expiration _ _ _ _ _ _

Montant _ _ _ _ , _ _ F Signature :

Câble-interface de communication Apple/Minitel

Cette liaison – décrite dans les numéros 27 et 28 de Pom's – est indispensable pour faire fonctionner les programmes suivants :

- **MinBas** pour Apple][+, //e, //e+, //c et //GS* : programme permettant l'enregistrement des écrans Minitel, la restitution à loisir hors réseau, le stockage et/ou l'impression de copies d'écran du Minitel, et aussi l'envoide textes ou messages sur un serveur, ces messages étant préparés à l'avance avec n'importe quel programme de traitement de textes. Programme publié dans le numéro 27 de Pom's.
- **Minitel/1** pour Macintosh : programme identique à MinBas pour Apple][, avec en plus un mini-éditeur de texte pour la préparation et le stockage des messages sans sortir du programme. Programme publié dansle numéro 27 de Pom's.
- **InterPom's** pour Apple][+, //e, //e+, //c et //GS* : programme de téléchargement entre Apple][et/ou Apple][et Macintosh. Ce logiciel autorise la transmission de n'importe quel type de fichier (système, texte, binaire, Basic...) en utilisant le Modem du Minitel. Programme publié dans le numéro 28 de Pom's.
- **InterPom's** pour Macintosh : même programme, mais version Macintosh. Programme publié dans le numéro 28 de Pom's.
- **TPom's** pour Apple][+, //e, //e+, //c et //GS** : programme de récupération de l'annuaire téléphonique sous la forme de fichiers texte. Voir page 53 de ce numéro.
- **TPom's** pour Macintosh : Identique à la version Apple][. Voir page 42 de ce numéro.

* ce programme est prévu pour fonctionner avec une carte Super Série Apple ; le port série intégré de l'Apple IIGS ne convient pas. Toutefois, si vous désirez utiliser le port intégré afin d'y connecter le Minitel pour d'autres applications, commandez un câble pour Macintosh Plus.

** sur un Apple IIGS, TPom's fonctionne indifféremment avec le port série intégré ou la carte Super Série Apple. Pour connecter le port intégré du IIGS, utilisez un câble pour Macintosh Plus.

Je désire recevoir :

câble Minitel/Apple][+, //e, //e+, IIGS avec SSC	____	à 225,00 F	_____
câble Minitel/Apple //c	____	à 225,00 F	_____
câble Minitel/Macintosh 128 & 512Ko	____	à 225,00 F	_____
câble Minitel/Macintosh Plus, IIGS port intégré	____	à 225,00 F	_____

* pour les envois par avion, ajoutez 15 F par câble.

Envoyez ce bon et votre règlement à : **Éditions MEV – 12, rue d'Anjou – 78000 Versailles**

Nom : _____

Adresse : _____

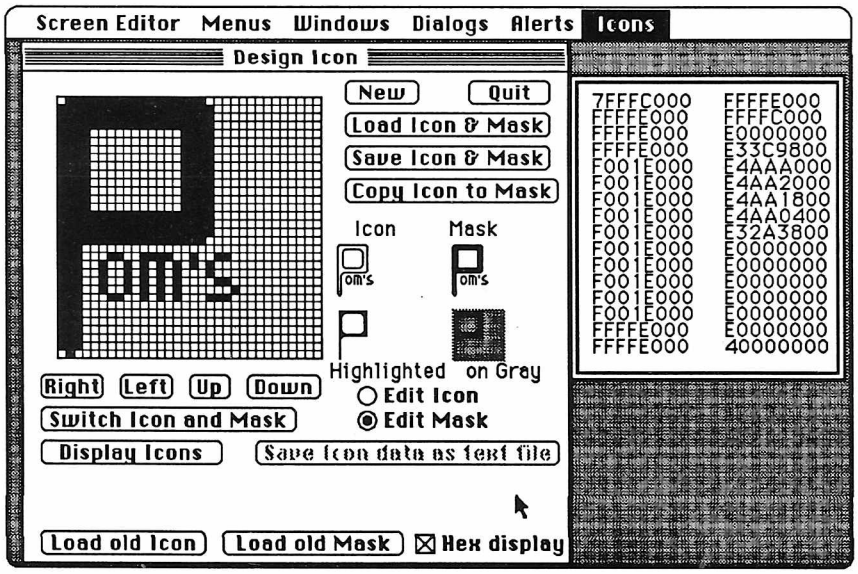
Règlement par : Carte Bleue/VISA ☐ Chèque bancaire ☐ Chèque postal ☐ Mandat ☐

numéro de la carte _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ date d'expiration _ _ _ _ _ _

Montant _ _ _ _ , _ _ F Signature :

Disquette Macintosh numéro 30

En plus du programme **TPom's** et des fichers MDS 68000 utilisés pour sa réalisation, la disquette Pom's 'Mac 30' contient l'application **Screen Editor**. Il s'agit d'un très puissant éditeur permettant la mise au point de menus, icônes, fenêtres d'alerte, 'boîtes à dialogue' et fenêtres de tous types. De plus, vous trouverez sur la disquette deux superbes images Mac Paint.



Dépôt légal : mai 1987 - N° d'impression : 788964. Imprimerie Berger-Levrault, Nancy.

SOYEZ BON AVEC VOTRE ORDINATEUR,

INSCRIVEZ-LE AU CENTRE DE MAINTENANCE DES ORDINATEURS

POUR RECEVOIR IMMÉDIATEMENT VOTRE CARTE D'ADHÉRENT, les privilèges auxquels vous avez droit, une documentation complète et les coordonnées du centre de maintenance le plus proche de vous. Cette inscription est gratuite. Votre centre de maintenance vous adresse régulièrement : une information sur votre ordinateur, logiciels et conseils d'entretien destinés à prolonger et à embellir la vie de votre ordinateur. Lorsque votre ordinateur est souffrant, VOUS L'APPORTEZ AU CENTRE DE MAINTENANCE ARAMIS. IL EST CONFIÉ 10 JOURS AU LABORATOIRE CENTRAL ARAMIS ET VOUS EST RENDU EN BONNE SANTÉ. ENTANT QU'ADHÉRENT, VOUS BÉNÉFICIEZ ENTRE AUTRES, DE CONDITIONS PRIVILÉGIÉES, DE PRIX D'INTERVENTION PRÉFÉRENTIELS, DE DÉLAIS RAPIDES. Les centres de maintenance ARAMIS peuvent intervenir et secourir rapidement votre ordinateur, quelle que soit la ville où vous résidez. Les centres accueillent les ordinateurs, imprimantes, écrans AMSTRAD, APPLE, THOMSON, COMMODORE, ZÉNITH, SINCLAIR, IBM et compatibles... Si votre ordinateur est en panne actuellement, appelez par le téléphone vert ci-dessous.

ARAMIS
GROUPE MIS

Demandez votre carte d'adhérent, c'est gratuit et sans engagement au Centre de maintenance ARAMIS. Allée des Frênes - 60290 RANTIGNY

NUMERO VERT APPEL GRATUIT (05) 1.2.4.8.16.

☐ Mme ☐ Mlle ☐ M. Nom : _____ Société : _____

Adresse : _____ Tél. : _____ ☐ Bureau ☐ Domicile

Êtes-vous : ☐ Particulier ☐ Société ☐ Distributeur – Profession : _____ Ou ☐ Étudiant

Marque et modèle de votre ordinateur : _____ Usage ☐ Familial ☐ Professionnel – Date de 1re mise en service : _____

COUPON A RETOURNER AU CENTRE DE MAINTENANCE ARAMIS - ALLÉE DES FRÊNES - 60290 RANTIGNY.

POM

MARINE